Global Optimisation for Energy Systems

Ksenia Bestuzheva

October 2018

A thesis submitted for the degree of Doctor of Philosophy of The Australian National University

Except where otherwise stated, this thesis is my own original work.

Ksenia Bestuzheva
October 2018

Abstract

The goal of global optimisation is to find globally optimal solutions, avoiding local optima and other stationary points. The aim of this thesis is to provide more efficient global optimisation tools for energy systems planning and operation. Due to the ongoing increasing of complexity and decentralisation of power systems, the use of advanced mathematical techniques that produce reliable solutions becomes necessary. The task of developing such methods is complicated by the fact that most energy-related problems are nonconvex due to the nonlinear Alternating Current Power Flow equations and the existence of discrete elements.

In some cases, the computational challenges arising from the presence of non-convexities can be tackled by relaxing the definition of convexity and identifying classes of problems that can be solved to global optimality by polynomial time algorithms. One such property is known as invexity and is defined by every stationary point of a problem being a global optimum. This thesis investigates how the relation between the objective function and the structure of the feasible set is connected to invexity and presents necessary conditions for invexity in the general case and necessary and sufficient conditions for problems with two degrees of freedom.

However, nonconvex problems often do not possess any provable convenient properties, and specialised methods are necessary for providing global optimality guarantees. A widely used technique is solving convex relaxations in order to find a bound on the optimal solution. Semidefinite Programming relaxations can provide good quality bounds, but they suffer from a lack of scalability. We tackle this issue by proposing an algorithm that combines decomposition and linearisation approaches.

In addition to continuous non-convexities, many problems in Energy Systems model discrete decisions and are expressed as mixed-integer nonlinear programs (MINLPs). The formulation of a MINLP is of significant importance since it affects the quality of dual bounds. In this thesis we investigate algebraic characterisations of on/off constraints and develop a strengthened version of the Quadratic Convex relaxation of the Optimal Transmission Switching problem.

All presented methods were implemented in mathematical modelling and optimisation frameworks PowerTools and Gravity.

Contents

Acknowledgements

First of all, I would like to thank my supervisory panel.

I am thankful to Dr. Hassan Hijazi for undertaking this journey with me, showing me interesting problems to work on and being an invaluable source of help on all aspects of the research process. Starting from the day when you welcomed me into the world of optimisation by placing a pile of papers in front of me (as I was sitting in the office wide-eyed, still confused by moving to another continent), continuing with the exciting discussions of the research lying ahead and sharing the hardships of model debugging and proof verifying, you have been a great mentor. You have been an example of scientific rigor and curiosity - the qualities that I will continue striving for in my further research career.

I am grateful to Prof. Sylvie Thiebaux for watching over the progress of my thesis and giving me valuable strategic advice. I greatly appreciate your guidance and support; despite you being a busy person, I have always felt that I could rely on you if I found myself in any difficult situation. As a passionate scientist and energetic and wise leader, you are an inspiration for me.

Many thanks also to Prof. Markus Hegland for being the mathematical sciences expert for this project, ready to help with any difficult proofs, and for inviting me to the group seminars where I could further extend my knowledge of mathematics. I enjoyed the atmosphere of genuine curiosity that you have created and encouraged at those seminars and will do my best to carry this attitude with me.

I gratefully acknowledge the funding provided by Data61, CSIRO for this work.

I would like to thank Dr. Carleton Coffrin for his contribution to the work on the Quadratic Convex relaxation of Optimal Transmission Switching, as well as others who were part of the optimisation team at that time for giving valuable feedback at the bi-annual presentations and creating a fun work environment (foosball!). I thank Dr. John Pye for letting me know about this Ph.D. program.

I express my deep appreciation to my mother for being with me through all the ups and downs of this journey, far in distance and time zones yet always close in heart; my grandmother for her loving and cheering presence on Skype; my grandfather for his support, and other family members. A special thanks to my friends both old and new for being great company, sharing passions, struggles or simply lunches and having conversations that were interesting, or heart-felt, or eye-opening, or funny or all those things at once.

Finally, I greatly appreciate the wonderful environment that Australia has been for me.

Wherever my life path takes me, I will always remember with warmth Australia's breathtaking beauty and friendly people.

Chapter 1

Introduction

The complexity of optimisation problems depends on the structure of the set of feasible solutions and the properties of the objective function. One crucial characteristic is convexity. Along with some mild nondegeneracy assumptions, convexity enables solving optimisation problems to global optimality or proving infeasibility with the use of polynomial time algorithms such as, for example, the widely used interior point methods.

However, many practical applications involve nonconvex constraints and objective functions, as well as discrete decisions. Among such application areas are energy systems, where the ongoing decentralisation and incorporation of renewable energy sources require solving challenging optimisation problems. Moreover, in energy systems the reliability of solutions is of critical importance in order to avoid system failures, and the requirements for computational efficiency are often high due to large problem size and the need to solve problems repeatedly in real time. This creates the need for specialised methods that do not rely on convexity of the problem for providing globally optimal solutions.

Various techniques have been developed for this purpose. Inexact approaches such as approximations and heuristics might be able to find approximated global optima efficiently, but their applicability is limited when the reliability of the solutions is crucial. This has motivated research on techniques that provide provable optimality and feasibility guarantees.

This thesis contributes to answering the following research question: given a nonconvex optimisation problem, how can we provide provable conclusions on its global optimum? In particular, how to show that the problem is infeasible or, given a feasible local optimal solution, prove its global optimality or evaluate the gap between this solution and the global optimum?

This work is focused on improving the performance of exact global optimisation methods as well as the quality of optimality guarantees provided. The goal is to extend the practical applicability of these methods to larger and more complex problems in energy systems.

In some special cases, it can be shown that every stationary point is the global optimum, and in the first part of the thesis we investigate new ways of identifying such cases. For problems that do not possess any known properties that make such a proof possible, convex relaxations are widely used. The second part of the thesis studies Semidefinite Programming relaxations of continuous problems and develops an algorithm that employs decomposition and linearisation in order to improve the scalability. Finally, we consider the case when the problem also involves discrete variables and on/off constraints, and the third part of the thesis presents a new formulation of quadratic on/off constraints and investigates ways of strengthening the Quadratic Convex relaxation [100] of the Optimal Transmission Switching problem in power systems.

Generalised Convexity

Under constraint qualifications [188] Karush-Kuhn-Tucker (KKT) conditions become both necessary and sufficient for global optimality in the case of convex problems [25]. Convexity, however, is not necessary in order for this property, known as KT-invexity, to hold and therefore can be generalised.

Since nonlinear solvers such as, for example, the interior-point solver Ipopt [187], provide polynomial time algorithms that do not in general find the global optimum but are guaranteed to converge to KKT points, KT-invexity enables their use for finding global optimal solutions. Moreover, identifying KT-invex subregions can improve the performance of spatial branch-and-bound algorithms.

Well known generalisations of convex functions include pseudo- and quasiconvex functions, and relaxations of the convexity property of optimisation problems have been proposed based on these notions. However, similarly to convexity, these conditions are not necessary for KT-invexity and there exist problems whose behaviour suggests KT-invexity, which are defined by nonquasi- and nonpseudoconvex functions. To gain more insight into the KT-invexity of such problems, one needs to consider the relations between the constraints and the objective function of the problem.

The conditions for KT-invexity that exist in the literature take these relations into account but, to the best of our knowledge, do not provide any clear procedure for identifying KT-invex problems. Therefore, the lack of algorithmically verifiable conditions still remains a major limitation of the invexity theory. We are addressing this by studying the behaviour of the objective function on the boundary of the feasible set and using it to identify classes of problems that are provably KT-invex. We define a new property which we call “boundary-invexity”, which ensures that certain structures that introduce multiple local minima are absent from the boundary of the feasible set. We prove that boundary-invexity is a necessary condition for KT-invexity in the general n-dimensional case and a sufficient condition in the case of problems with two degrees of freedom.

Boundary-invexity of a problem can be verified by solving several smaller subproblems, one for each non-convex constraint. Although in general these subproblems are still NP-hard, they can be more computationally tractable than the original problem due to smaller size and, in some cases, special structure.

KT-invexity of Optimal Power Flow The OPF problem has been proven to be NP-hard in the general case even for acyclic graphs [186, 124]. However, there is empirical evidence suggesting that OPF is KT-invex under some realistic assumptions on the parameter values and variable bounds, but a formal proof is necessary in order to guarantee global optimality of KKT points. As an example, we study invexity properties of a class of OPF problems with two degrees of freedom defined on networks that consist of two buses connected by one line. We show that under some realistic assumptions on the parameters, these problems are KT-invex.

Semidefinite Programming Relaxations for OPF

Many optimisation problems are not KT-invex or cannot be proven to be KT-invex using known methods, and efficient global optimisation algorithms are necessary in order to prove global optimality of their solutions. In such cases convex relaxations are widely used to evaluate the gap between a local optimal solution of a nonconvex problem and the global optimum.

Semidefinite programming (SDP) relaxations have been shown to yield tight bounds for the Optimal Power Flow (OPF) problem [122]. However, the scalability of the state of the art SDP solvers is limited.

Decomposition techniques such as constraint generation and exploiting sparsity of the graphs have been successfully applied to improve the efficiency of solution algorithms. Building upon these results, we develop a linear cut generation algorithm which avoids adding the computationally challenging SDP constraints to the model. First, we apply tree decomposition to the sparsity pattern graph in order to obtain an equivalent formulation of the SDP problem written in terms of smaller matrices. Then we investigate the impact of different linear cuts on the search space, aiming to improve the reliability and efficiency of the approach. The notion of the "deepest valid cut" with respect to the Euclidean norm is introduced. In practice these cuts are obtainable by solving a projection subproblem. Using additional information about the problem such as which constraints tend to be active, we improve the SDP condition verification process, which allows us to detect more violated constraints and improve the gap yielded by our approach.

The resulting dynamic cut generation algorithm is applied to the Semidefinite Programming relaxation of the OPF problem and is shown to improve the robustness compared to standard SDP approaches.

Another option is to solve SDP problems by replacing the positive semidefiniteness matrix constraints by their nonlinear equivalents, thus converting an SDP problem into a polynomial optimisation problem which can be solved by efficient nonlinear programming algorithms. In a relaxation proposed by Hijazi et al. [99] only those constraints that correspond to tree decomposition bags (sets of nodes of the original graph that correspond to nodes of the tree it is mapped into) of size 3 were added to the model. In this work we extend this approach by adding constraints that correspond to all principal minors of size 3 of tree decomposition bags. The proposed formulation is shown to be more computationally efficient than the standard sparse SDP formulation and yield the same bounds as the full SDP relaxation on medium-sized OPF test cases.

On/off Constraints and Convex Relaxations of Optimal Transmission Switching

In Mixed-Integer Nonlinear Programs (MINLPs) the requirements on variables' integrality are an additional source of non-convexity. For such problems the formulation plays a particularly important role because it affects the quality of continuous relaxations which are used by branch and bound algorithms.

In this thesis we study constraints that are included into the model when the corresponding binary variable is equal to one and are ignored otherwise. Such constraints are referred to as on/off or disjunctive. In order to pass an on/off constraint to a MINLP solver, one has to find its algebraic formulation. Importantly, its continuous relaxation should be convex in order for the problem to be solvable to global optimality by efficient convex MINLP solvers which make use of the convexity of the continuous relaxation of the problem. Formulations that are written in the space of original variables and yield tight continuous relaxations typically lead to improved performance and thus are of interest.

We extend the perspective function based approach presented by Hijazi et al. [98] to non-monotone constraints by using the inverse of a function. Considering the feasible set of a two-dimensional quadratic on/off constraint as a union of two disjoint sets, we construct its algebraic formulation by finding the convex hull of those sets. The definition of a convex hull implies that such a formulation results in the tightest possible continuous relaxation. Moreover, our characterisation does not involve any additional variables. To avoid numerical issues arising from the nondifferentiability of perspective functions, we generate linear outer approximations of the convex hull.

New quadratic outer approximations of trigonometric functions are proposed, given that the function arguments have such bounds that ensure that the function is either convex or concave in the feasible set.

As an application, we study the Quadratic Convex (QC) relaxation of the Optimal Transmission Switching problem (OTS), which is obtained from OPF by allowing line switching. We apply the new outer approximations and the convex hull formulation to construct improved relaxations of trigonometric on/off constraints in QC-OTS. Our experiments indicate that the convex hull formulation reduces the average time required for solving medium-sized QC-OTS instances. We further tighten the QC relaxation by utilising our new quadratic relaxations of trigonometric constraints, applying bound propagation, adding valid cuts and improving the calculation of the big-M constants. The strengthened relaxation closes the gap on 5 out of 23 test instances compared to the standard QC formulation.

Main Contributions

Here we provide a brief summary of the main contributions of the thesis:

- An algorithmically verifiable necessary and sufficient condition for KT-invexity of problems with two degrees of freedom;
- An algorithmically verifiable necessary condition for KT-invexity of problems with no restrictions on the number of degrees of freedom;
- A proof of sufficiency of global optimality on the boundary of the feasible set of a continuous nonconvex problem for global optimality in the interior;
- An algorithm for dynamic generation of linear Semidefinite Programming cuts with an application to the Optimal Power Flow problem;
- A compact characterisation of the convex hull of a two-dimensional quadratic on/off constraint;
- New quadratic outer approximations of trigonometric functions;
- A strengthened version of the Convex Quadratic relaxation of the Optimal Transmission Switching problem.

Structure of the Thesis

The structure of the thesis is as follows:

Chapter 2 provides a summary of basic optimisation concepts and reviews the application area of energy systems. It describes the Optimal Power Flow and Optimal Transmission Switching problems which motivated the theoretical research in this thesis and restates the formulations of their state of the art convex relaxations. The benchmarks that are used in the computational experiments are discussed.

In Chapter 3 we present the new conditions for KT-invexity and the necessity and sufficiency proofs. The connection between global optimality on the boundary and in the interior of the feasible set is established. The sufficient conditions are applied to prove KT-invexity of OPF problems with two degrees of freedom given mild assumptions on the variables' bounds.

Chapter 4 studies convex relaxations of nonconvex problems. We develop a dynamic cut generation algorithm based on decomposition and linearisation of SDP constraints and apply it to the Optimal Power Flow problem.

Chapter 5 continues the work on convex relaxations, focusing on mixed-integer programs. An algebraic formulation of a disjunctive constraint is proposed and new quadratic relaxations of constraints with trigonometric functions are built. Together with some other techniques these results are used to tighten the Convex Quadratic relaxation of the Optimal Transmission Switching problem.

Chapter 6 summarises our findings and discusses directions for future research.

Notations and Basic Definitions

Throughout the thesis, we will use the following notation:

∂S	boundary of a set S,
x_i	ith component of vector $\mathbf{x}$,
$f'_{x_i} = \frac{\partial f}{\partial x_i}$	partial derivative of f with respect to x_i,
$\mathbf{x} \cdot \mathbf{y}$	dot product of vectors $\mathbf{x}$ and $\mathbf{y}$,
$\mathbf{x}^T$	transpose of vector $\mathbf{x}$,
$\overline{AB}$	a segment between points A and B,
$2\mathbb{N}$, $2\mathbb{N}+1$	the sets of even and odd numbers,
$f'_-(x), f'_+(x)$	left and right derivatives of f,
$\boldsymbol{i}$	imaginary unit,
$\Re(x)$	real part of a complex number x,
$\Im(x)$	imaginary part of a complex number x,
x^*	conjugate of a complex number x,
$\|\| \cdot \|\|$	Frobenius norm of a matrix or l^2-norm of a vector,
$\| \cdot \|$	cardinality of a set or l^1-norm of a scalar number,
$\mathrm{conv}(S)$	convex hull of a set S.

Bold italic font will be used for constants and bold font will denote vectors.

Given a differentiable function f, $\frac{\partial f(\mathbf{x})}{\partial \mathbf{u}}$ will denote the directional derivative of f along vector $\mathbf{u}$ which is defined as:

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{u}} = (\nabla f(\mathbf{x}))^T \cdot \mathbf{u}. \text{ [192]}$$

$\mathrm{sign}(x)$ stands for the sign function:

$$\mathrm{sign}(x) = \begin{cases} 1 & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0. \end{cases}$$

Chapter 2

Background

This chapter provides background material that is relevant for the work presented in the subsequent chapters of this thesis. First, in Section 2.1 we restate some basic definitions and results from optimisation theory. Section 2.2 concentrates on the nonconvex case and gives an overview of the methods that aim at providing global optimality guarantees.

Going on to the practical application, Section 2.3 presents the Alternating Current Optimal Power Flow (AC-OPF) problem and discusses the optimisation techniques that have been applied to it. In Section 2.4 we review convex relaxations that have been proposed for the AC-OPF problem. Sections 2.5 and 2.6 consider a mixed-integer extension of AC-OPF known as Optimal Transmission Switching, present its formulation and convex relaxations.

The chapter is concluded by Section 2.7 which describes the benchmarks used for our computational experiments.

2.1 Optimisation Basics

In this section we recall some basic definitions and classical results in optimisation.

Consider a constrained nonlinear optimisation problem in the following general form:

$$\begin{aligned} \min\ & f(\mathbf{x}) \\ \text{s.t. } & g_i(\mathbf{x}) \leqslant 0,\ i = 1, \dots, m, \\ & h_j(\mathbf{x}) = 0,\ j = 1, \dots, k, \\ & \mathbf{x} \in \mathbb{R}^n, \end{aligned} \tag{NLP_G}$$

where f, g_i, $i = 1 \dots, m$ and h_j, $j = 1, \dots, k$ are twice differentiable functions. Some of the results discussed here can be extended to the nondifferentiable case, however, this is not the focus of this work. Let F denote the feasible set of (NLP_G):

$$F = \{\mathbf{x} \in \mathbb{R}^n \mid g_i(\mathbf{x}) \leqslant 0\ \forall i = 1, \dots, m,\ h_j(\mathbf{x}) = 0\ \forall j = 1, \dots, k\}.$$

First, let us provide formal definitions of the solutions of optimisation problems. Finding a global optimum is most preferable:

Definition 2.1. *[150] A point* $\mathbf{x}^* \in \mathbb{R}^n$ *is a global minimiser for* (NLP_G) *if* $\mathbf{x}^* \in F$ *and* $f(\mathbf{x}) \geqslant f(\mathbf{x}^*)$ *for* $\mathbf{x} \in F$.

However, in many cases most algorithms cannot guarantee global optimality and provably converge to a local optimum:

Definition 2.2. *[150] A point* $\mathbf{x}^* \in \mathbb{R}^n$ *is a local minimiser for* (NLP_G) *if* $\mathbf{x}^* \in F$ *and there exists a neighbourhood* $N(\mathbf{x}^*)$ *such that* $f(\mathbf{x}) \geqslant f(\mathbf{x}^*)$ *for* $\mathbf{x} \in N(\mathbf{x}^*) \cap F$.

If the inequality in the above definition is strict for all points in the neighbourhood except for $\mathbf{x}^*$ itself, the minimiser is called strict:

Definition 2.3. *[150] A point* $\mathbf{x}^* \in \mathbb{R}^n$ *is a strict local minimiser for* (NLP_G) *if* $\mathbf{x}^* \in F$ *and there is a neighbourhood* $N(\mathbf{x}^*)$ *such that* $f(\mathbf{x}) > f(\mathbf{x}^*)$ *for* $\mathbf{x} \in N(\mathbf{x}^*) \cap F \setminus \mathbf{x}^*$.

First-order optimality conditions can be expressed in terms of constraint gradient vectors at a given point:

Definition 2.4. *[109, 117] A solution* $\mathbf{x}^*$ *of problem* (NLP_G) *is said to satisfy Karush-Kuhn-Tucker (KKT) conditions if there exist constants* μ_i $(i = 1, \ldots, m)$ *and* ν_j $(j = 1, \ldots, k)$*, called Lagrange multipliers, such that*

$$\nabla f(\mathbf{x}^*) = -\sum_{i=1}^{m} \mu_i \nabla g_i(\mathbf{x}^*) - \sum_{j=1}^{k} \nu_j \nabla h_j(\mathbf{x}^*), \tag{2.1}$$

$$g_i(\mathbf{x}^*) \leqslant 0, \;\; \forall i = 1, \ldots, m, \tag{2.2}$$

$$h_j(\mathbf{x}^*) = 0, \;\; \forall j = 1, \ldots, k, \tag{2.3}$$

$$\mu_i \geqslant 0, \;\; \forall i = 1, \ldots, m, \tag{2.4}$$

$$\mu_i g_i(\mathbf{x}) = 0, \;\; \forall i = 1, \ldots, m. \tag{2.5}$$

In the general nonconvex case, KKT conditions are necessary for a local optimum if constraint qualifications are satisfied. A widely used constraint qualification which we will utilise in this work is the linear independence constraint qualification:

Definition 2.5. *[150] A point* $\mathbf{x}^*$ *is said to satisfy the linear independence constraint qualification (LICQ) if the set of active constraint gradients* $\{\nabla g_i(\mathbf{x}^*), i \in A(\mathbf{x}^*),\ \nabla h_j(\mathbf{x}^*),\ j = 1, \ldots, k\}$ *is linearly independent, where* $A(\mathbf{x}^*)$ *be the set of indices of all active inequality constraints at point* $\mathbf{x}^*$.

Further information on constraint qualifications can be found in the book by Nocedal and Wright [150].

KKT conditions play an important role in optimisation and serve as the basis for many methods.

Generally, first order conditions do not guarantee even local optimality. Second derivatives have to be considered in order to distinguish between local minima and other types of stationary points such as local maxima and saddle points. Second derivatives provide information about the local convexity/concavity of functions, and their role is to determine the behaviour of functions in the "undecided" feasible directions $\mathbf{w}$ where $\mathbf{w}^T \nabla f(\mathbf{x}^*) = 0$.

The set of directions that need to be considered in order to define second order conditions is known as a critical cone:

Definition 2.6. *[150] Given a KKT point* $\mathbf{x}^*$ *of problem* (NLP_G) *and corresponding Lagrange multiplier vectors* μ, ν, *a critical cone* $C(\mathbf{x}^*, \mu, \nu)$ *is defined as a set of vectors* $\mathbf{w}$ *such that:*

$$\begin{cases} (\nabla g_i(\mathbf{x}^*))^T \cdot \mathbf{w} = 0 \ \forall i \in A(\mathbf{x}^*) \text{ with } \mu_i > 0, \\ (\nabla g_i(\mathbf{x}^*))^T \cdot \mathbf{w} \leqslant 0 \ \forall i \in A(\mathbf{x}^*) \text{ with } \mu_i = 0, \\ (\nabla h_j(\mathbf{x}^*))^T \cdot \mathbf{w} = 0 \ \forall j = 1, \ldots, k, \end{cases}$$

where $A(\mathbf{x}^*)$ *is the set of indices of all active inequality constraints at point* $\mathbf{x}^*$.

It can be observed that KKT conditions imply that $\mathbf{w}^T \nabla f(\mathbf{x}^*) = 0$ for all $\mathbf{w}$ in the critical cone. Therefore, critical cone contains undecided directions. Using this definition, we can write the second order conditions:

Theorem 2.1. *[150](Second-order sufficient conditions) Let* $\mathbf{x}^*$ *be a KKT point for problem* (NLP_G) *with Lagrange multiplier vectors* μ, ν. *Suppose that*

$$\mathbf{w}^T \nabla^2_{\mathbf{x}} L(\mathbf{x}^*, \mu, \nu) \mathbf{w} > 0 \ \forall \mathbf{w} \in C(\mathbf{x}^*, \mu, \nu), \ \mathbf{w} \neq 0,$$

where $L(\mathbf{x}, \mu, \nu) = f(\mathbf{x}) + \sum_{i=1}^{m} \mu_i g_i(\mathbf{x}) + \sum_{j=1}^{k} \nu_j h_j(\mathbf{x})$ *is the Lagrangian function.*

Then $\mathbf{x}^*$ *is a strict local minimum in* (NLP_G).

Convex optimisation If f, g_i, $i = 1, \ldots, m$ and h_j, $j = 1, \ldots, k$ are convex, (NLP_G) is a convex optimisation problem.

For convex problems, KKT conditions (2.1)-(2.5) become both necessary and sufficient for a global optimum [25]. Other important results proven for convex problems include weak and, under constraint qualifications, strong duality, convergence and convergence rates of interior-point algorithms.

Since polynomial time algorithms exist that guarantee convergence to KKT points, convex problems are solvable to global optimality in polynomial time.

2.2 Global Optimisation Methods

If a problem is nonconvex, finding the global optimum becomes much more challenging. There are several ways of approaching this task.

2.2.1 Upper bounding methods

Local optimisation algorithms can be applied to nonconvex problems, but with significant limitations. In the case where certain generalised convexity properties such as, for example, pseudo- and quasi- convexity or invexity [131, 133], can be proven, some of these methods guarantee global optimality, but in the general case convergence even to a local optimum is not guaranteed. For example, such methods might converge to a saddle point or report local infeasibility. Since they cannot guarantee infeasibility of the whole problem, no informative conclusions can be made in the latter case.

Despite these issues, convex optimisation methods are widely used for evaluating an upper bound on the optimal solution of a nonconvex problem.

2.2.2 Convex relaxations

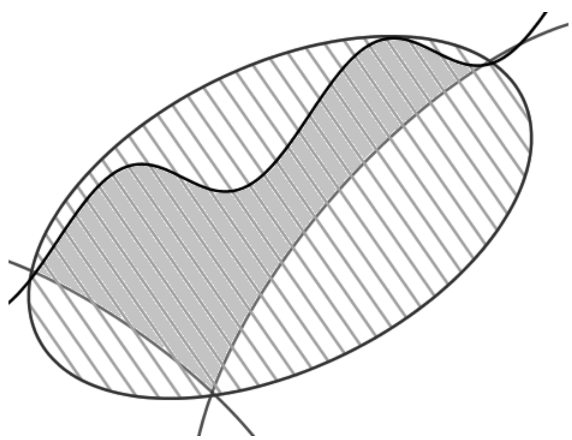


Figure 2.1: Example of a convex relaxation: the nonconvex set is shown in grey colour, the hatched region represents the relaxation

By definition, a convex relaxation of a nonconvex optimisation problem is a problem of optimising the same objective function over a convex set that includes the original feasible set. Therefore, the optimal objective function value of a relaxation is guaranteed to be less than or equal to the global optimum of the original problem [25] (or greater than or equal in the case of a maximisation problem). Such a value is called a lower bound. At the same time, a local optimum of the original (minimisation) problem is known to be greater than or equal to the global optimum and is referred to as an upper bound. Both values can be computed in polynomial time and can be used to evaluate how far from the global optimum the objective function value at a local solution is. If the lower bound equals the upper bound, the globally optimal value has been found. Moreover, if the relaxation is proven to be infeasible, so is the original problem.

The closer the relaxation is to the original problem, the better optimality guarantees it provides. This has motivated research on convexification techniques that lead to tighter formulations.

2.2.3 Spatial branch and bound

The idea of spatial branch and bound algorithms [172, 180] is to systematically explore the feasible set by dividing it into smaller subregions. To divide the feasible region, a variable is chosen and its domain is separated into two parts, thus generating two subsets of the original set. This step is called branching. When exploring a subregion, local optimisation techniques, heuristics, convex relaxations and other methods are applied in order to obtain upper and lower bounds on the optimal solution. This process is known as bounding. By repeating these steps, upper and lower bounds are improved until a specified tolerance is reached.

Spatial branch and bound algorithms converge to the global optimum, although most often at a high computational cost.

2.2.4 Mixed-integer programming

In mixed-integer programming the integrality requirements on some variables present an additional computational challenge. Many methods that are commonly applied to solve such problems belong to the family of mixed-integer branch and bound methods [118, 50]. The idea is similar to that of spatial branch and bound with the following differences:

- the branching is done by choosing different values of the discrete variables,
- convex relaxations of nonconvex sets that are used in spatial branch and bound are replaced by continuous relaxations of mixed-integer sets.

A more detailed review of mixed-integer programming methods can be found in Section 5.2.

For solving nonconvex mixed-integer nonlinear programs, spatial and mixed-integer branch and bound can be combined in one algorithm.

2.3 The Optimal Power Flow Problem

Optimal Power Flow (OPF) is a problem of finding the generation dispatch that minimises a given objective, usually the operating cost. This problem is also known as Alternating Current Optimal Power Flow (AC-OPF) since it deals with alternating current networks. It was first introduced by Carpentier [32] as an extension of the Economic Dispatch problem. OPF forms the basis of many energy related applications such as: Security Constrained Optimal Power Flow which takes contingencies into account [31]; Unit Commitment, the problem of optimal scheduling of generation units, including the on-off status of each generator, across multiple time periods [154]; Reactive Power Planning which seeks to optimally allocate reactive power sources [196, 65]; Optimal Transmission Switching which is obtained from OPF by allowing line activation/deactivation [59].

2.3.1 Solution methods

AC-OPF is a challenging optimisation problem due to the nonconvex nature of physical laws governing the processes in the network. It has been proven to be NP-hard in the general case [186, 124]. Moreover, many OPF-based applications include discrete control elements and therefore need to be modelled as mixed-integer nonlinear programs. This has prompted extensive research on optimisation techniques tailored for OPF.

Local optimisation approaches include the adaptations of such classical nonlinear programming algorithms as gradient methods [53, 156], Newton-Raphson [36, 161], sequential linear and quadratic programming [175, 176, 27] and interior-point methods [83, 183].

DC (direct current) approximations [162, 163] have been widely used to linearise the OPF problem. These approximations model alternating current networks, and the name is derived from the observation that the linearised constraints resemble direct current power flow equations. The DC model is computationally efficient and is a reasonably good approximation of AC power flows under normal operating conditions. However, it ignores reactive power and thus is not suitable for reactive power planning applications, and the key assumptions that ensure the accuracy of the model are often not satisfied [177]. A different linear approximation that takes into account reactive power and voltage magnitudes was proposed by Coffrin and Van Hentenryck [44].

While approximations are useful for some applications, there is a need for methods that can provide provable optimality and feasibility guarantees. Recently there has been a lot of interest in convex relaxations of OPF since they can efficiently produce lower bounds on solutions. When combined with local optimisation techniques they can be used to evaluate how far the objective function value at a local solution is from the global optimum. Convex relaxations are also used as part of spatial branch and bound algorithms [157, 82]. Relaxations of the OPF problem are discussed in Section 2.4 of this chapter.

2.3.2 Formulation

Let us restate the formulation of the OPF problem. Consider an electrical network with the set of buses (nodes) and set of lines (arcs) denoted as N and E respectively. The following parameters and variables describe the physical characteristics of the network:

Parameters:

$\boldsymbol{Y}_{ij} = \boldsymbol{g}_{ij} + \boldsymbol{i}\boldsymbol{b}_{ij}$	admittance of line $(i,j) \in E$, where the real part is conductance and the imaginary part is susceptance,
$\boldsymbol{s}^u_{ij}$	thermal limit of line $(i,j) \in E$,
$\boldsymbol{S}^d_i = \boldsymbol{p}^d_i + \boldsymbol{i}\boldsymbol{q}^d_i$	power demand at node $i \in N$, where the real part is active power and the imaginary part is reactive power (demand is assumed to be constant),
$\boldsymbol{c}_{0i},\ \boldsymbol{c}_{1i},\ \boldsymbol{c}_{2i}$	generation cost coefficients at node $i \in N$.

Variables:

$S_{ij} = p_{ij} + \boldsymbol{i}q_{ij}$	electric power along line $(i, j) \in E$, consisting of active and reactive power,
$V_i = v_i \angle \theta_i$	voltage at node $i \in N$ with magnitude v_i and phase angle θ_i,
$W_{ij} = w^R_{ij} + \boldsymbol{i}w^I_{ij}$	product of voltages at nodes $i \in N$ and $j \in N$,
w_i	squared voltage magnitude at node $i \in N$,
$\theta_{ij} = \theta_i - \theta_j$	phase angle difference between nodes $i \in N$ and $j \in N$,
$S^g_i = p^g_i + \boldsymbol{i}q^g_i$	power generation at node $i \in N$, consisting of active and reactive power.

Upper indices $(\cdot)^l$ and $(\cdot)^u$ denote lower and upper bounds on variables.

The OPF problem in complex form is given by Model 2.1:

Model 2.1: The Optimal Power Flow problem, complex form

variables for each $(i, j) \in E$:

S_{ij}, W_{ij}

variables for each $i \in N$:

$V_i,\ S^g_i$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i}(\Re(S^g_i))^2 + \boldsymbol{c}_{1i}\Re(S^g_i) + \boldsymbol{c}_{0i} \right) \tag{2.6a}$$

subject to:

$$\angle V_{\boldsymbol{r}} = 0 \tag{2.6b}$$

$$W_{ij} = V_i V_j^* \quad \forall (i, j) \in E \tag{2.6c}$$

$$S^g_i - \boldsymbol{S}^d_i = \sum_{(i,j) \in E} S_{ij} + \sum_{(j,i) \in E} S_{ij} \quad \forall i \in N \tag{2.6d}$$

$$S_{ij} = \boldsymbol{Y}^*_{ij} W_{ii} - \boldsymbol{Y}^*_{ij} W_{ij} \quad \forall (i, j) \in E \tag{2.6e}$$

$$|S_{ij}| \leqslant \boldsymbol{s}^{\boldsymbol{u}}_{ij} \quad \forall (i, j), (j, i) \in E \tag{2.6f}$$

$$-\tan(\boldsymbol{\theta}^{\boldsymbol{u}}_{ij})\Re(W_{ij}) \leqslant \Im(W_{ij}) \leqslant \tan(\boldsymbol{\theta}^{\boldsymbol{u}}_{ij})\Re(W_{ij}) \ \forall (i, j) \in E \tag{2.6g}$$

$$\Re(\boldsymbol{S}^{\boldsymbol{gl}}_i) \leqslant \Re(S^g_i) \leqslant \Re(\boldsymbol{S}^{\boldsymbol{gu}}_i) \quad \forall i \in N \tag{2.6h}$$

$$\Im(\boldsymbol{S}^{\boldsymbol{gl}}_i) \leqslant \Im(S^g_i) \leqslant \Im(\boldsymbol{S}^{\boldsymbol{gu}}_i) \quad \forall i \in N \tag{2.6i}$$

$$(\boldsymbol{v}^{\boldsymbol{l}}_i)^2 \leqslant |V_i|^2 \leqslant (\boldsymbol{v}^{\boldsymbol{u}}_i)^2 \quad \forall i \in N \tag{2.6j}$$

The Alternating Current Power Flow equations (2.6e) derived from Ohm's law together with Kirchhoff's Current law equations (2.6d) form the core of the OPF problem. Nonlinear equation (2.6e) is the source of non-convexity in the model. In addition to these physical constraints, OPF includes the operational constraints such as variable bounds and thermal limits ((2.6f)-(2.6j)) and the reference bus equation (2.6b). Unless specified otherwise, in all the models given here the lower and upper bounds on phase angle differences θ_{ij} are assumed to satisfy $-\pi/2 < \boldsymbol{\theta}^l_{ij} < \boldsymbol{\theta}^u_{ij} < \pi/2$ and to be symmetrical ($\boldsymbol{\theta}^l_{ij} = -\boldsymbol{\theta}^u_{ij}$).

Model 2.1 is often transformed into either the polar or rectangular real form. In both cases the apparent power S_{ij} is replaced by the active and reactive powers p_{ij}, q_{ij}. The polar formulation is obtained by rewriting the complex terms by using the polar form of complex numbers and separating the equations that correspond to real and imaginary parts of the power flows:

Model 2.2: The Optimal Power Flow problem, polar form

---

variables for each $(i,j) \in E$:

$p_{ij},\ q_{ij},\ \theta_{ij} \in [-\boldsymbol{\theta}^u_{ij},\ \boldsymbol{\theta}^u_{ij}]$

variables for each $i \in N$:

$v_i \in [\boldsymbol{v}^l_i,\ \boldsymbol{v}^u_i],\ \theta_i$

$p^g_i \in [\boldsymbol{p}^{\boldsymbol{gl}}_i,\ \boldsymbol{p}^{\boldsymbol{gu}}_i],\ q^g_i \in [\boldsymbol{q}^{\boldsymbol{gl}}_i,\ \boldsymbol{q}^{\boldsymbol{gu}}_i]$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i} (p^g_i)^2 + \boldsymbol{c}_{1i} p^g_i + \boldsymbol{c}_{0i} \right)$$

subject to:

$$\theta_{\boldsymbol{r}} = 0 \tag{2.7a}$$

$$\theta_{ij} = \theta_i - \theta_j \quad \forall (i,j) \in E \tag{2.7b}$$

$$p^g_i - \boldsymbol{p}^d_i = \sum_{(i,j) \in E} p_{ij} + \sum_{(j,i) \in E} p_{ij} \quad \forall i \in N \tag{2.7c}$$

$$q^g_i - \boldsymbol{q}^d_i = \sum_{(i,j) \in E} q_{ij} + \sum_{(j,i) \in E} q_{ij} \quad \forall i \in N \tag{2.7d}$$

$$p_{ij} = \mathbf{g}_{ij} v_i^2 - \mathbf{g}_{ij} v_i v_j \cos(\theta_{ij}) - \mathbf{b}_{ij} v_i v_j \sin(\theta_{ij}) \quad \forall (i,j) \in E \tag{2.7e}$$

$$q_{ij} = -\mathbf{b}_{ij} v_i^2 + \mathbf{b}_{ij} v_i v_j \cos(\theta_{ij}) - \mathbf{g}_{ij} v_i v_j \sin(\theta_{ij}) \quad \forall (i,j) \in E \tag{2.7f}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{s}^{\boldsymbol{u}}_{ij} \quad \forall (i,j) \in E \tag{2.7g}$$

---

where equations (2.7c), (2.7d) are equivalent to (2.6d), constraints (2.7e), (2.7f) capture Ohm's law (2.6e) and inequality (2.7g) enforces the thermal limits.

Alternatively, the voltages can be written using the rectangular form $V_i = v^R_i + \boldsymbol{i} v^I_i$ to obtain the rectangular formulation of OPF.

In practice, the Optimal Power Flow problem usually incorporates such parameters as transformers, shunts and line charging, which are omitted here for brevity purposes. All the models presented here can easily be extended to include those components.

2.4 Relaxations of OPF

In this section we will review the convex relaxations of the OPF problem. The main idea behind these formulations is to substitute the nonlinear terms in the power flow equations by auxiliary variables and then impose convex constraints on these variables that capture certain aspects of the behaviour of the nonconvex expressions they represent.

2.4.1 Semidefinite programming relaxations

A semidefinite programming (SDP) constraint has the form:

$$X \succeq 0, \tag{2.8}$$

requiring matrix X to be positive semidefinite (PSD). (2.8) describes a convex region. SDP constraints are often used for formulating convex relaxations of nonconvex continuous and combinatorial optimisation problems [144, 78]. The SDP relaxation of OPF was introduced by Bai et al. [8]. Although computationally expensive, the SDP relaxation has the advantage of good solution quality. It has been proven to be exact in some special cases [122, 173, 195, 123]. The limitations of the SDP relaxation in terms of exactness were shown in a work by Lesieutre et al. [125], where a three-bus example with nonzero optimality gap was given.

The SDP formulation If we let

$$W_{ij} = V_i V_j^*, \tag{2.9}$$

then it can be observed that nonzero V_i, V_j satisfying (2.9) exist if and only if the matrix

$$W = \begin{cases} W_{ii} = w_i \ \forall i \in N, \\ W_{ij} = w_{ij}^R + \boldsymbol{i} w_{ij}^I \ \forall \ (i,j) \in N \times N, \end{cases}$$

is positive semidefinite and has rank 1. This allows us to eliminate V from the formulation by replacing (2.9) with:

$$W \succeq 0,$$
$$\text{rank}(W) = 1$$

and rewriting the power flow equations (2.6e) in terms of variables w_i, w_{ij}^R and w_{ij}^I:

$$p_{ij} = \mathbf{g}_{ij} w_i - \mathbf{g}_{ij} w_{ij}^R - \mathbf{b}_{ij} w_{ij}^I \quad \forall (i,j) \in E,$$
$$q_{ij} = -\mathbf{b}_{ij} w_i + \mathbf{b}_{ij} w_{ij}^R - \mathbf{g}_{ij} w_{ij}^I \quad \forall (i,j) \in E.$$

In addition to constraint (2.9), the values of the original variables V_i, V_j need to satisfy the operational constraints. In the w-space the bounds on voltage magnitudes and phase angle differences are written as:

$$(\boldsymbol{v}_i^l)^2 \leqslant w_i \leqslant (\boldsymbol{v}_i^u)^2, \tag{2.10}$$

$$w_{ij}^R \tan(\boldsymbol{\theta}_{ij}^l) \leqslant w_{ij}^I \leqslant w_{ij}^R \tan(\boldsymbol{\theta}_{ij}^u). \tag{2.11}$$

Bounds on the new variables w_{ij}^R and w_{ij}^I for all $(i,j) \in E$ are as follows:

$$\boldsymbol{v}_i^l \boldsymbol{v}_j^l \cos(\boldsymbol{\theta}_{ij}^u) \leqslant w_{ij}^R \leqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u, \tag{2.12}$$

$$-\boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}_{ij}^u) \leqslant w_{ij}^I \leqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}_{ij}^u). \tag{2.13}$$

In the case when the line (i,j) does not exist (i.e. $(i,j) \in \{N \times N\} \backslash E$), the lower and upper bounds on the phase angle difference are instead defined by the sum of bounds (lower or upper, respectively) on the shortest path connecting nodes i and j. In practice, in order to avoid finding shortest paths between all nonadjacent nodes in the network, these are usually relaxed and replaced by the sum of bounds of all existing lines. Let $\boldsymbol{M}^u, \boldsymbol{M}^l$ denote the relaxed bound:

$$\boldsymbol{M}^u = \sum_{(i,j) \in E} \boldsymbol{\theta}_{ij}^u, \tag{2.14}$$

$$\boldsymbol{M}^l = -\boldsymbol{M}^u. \tag{2.15}$$

Using this notation, we can now write the bounds on the remaining w_{ij}^R, w_{ij}^I variables:

$$w_{ij}^R \leqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u, \tag{2.16}$$

$$\begin{cases} w_{ij}^R \geqslant \boldsymbol{v}_i^l \boldsymbol{v}_j^l \cos(\boldsymbol{M}^u) \text{ if } \boldsymbol{M}^u < \pi/2, \\ w_{ij}^R \geqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u \cos(\boldsymbol{M}^u) \text{ if } \pi/2 \leqslant \boldsymbol{M}^u < \pi, \\ w_{ij}^R \geqslant -\boldsymbol{v}_i^u \boldsymbol{v}_j^u \text{ if } \pi \leqslant \boldsymbol{M}^u, \end{cases} \tag{2.17}$$

$$\begin{cases} -\boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}_{ij}^u) \leqslant w_{ij}^I \leqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}^u) \text{ if } \boldsymbol{M}^u < \pi/2, \\ -\boldsymbol{v}_i^u \boldsymbol{v}_j^u \leqslant w_{ij}^I \leqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u \text{ if } \pi/2 \leqslant \boldsymbol{M}^u. \end{cases} \tag{2.18}$$

The bounds calculated according to formulas (2.12)-(2.18) will be denoted as $(\boldsymbol{w}_{ij}^R)^l$, $(\boldsymbol{w}_{ij}^R)^u$, $(\boldsymbol{w}_{ij}^I)^l$ and $(\boldsymbol{w}_{ij}^I)^u$.

Obtaining the convex relaxation The rank 1 constraint captures all the non-convexity in OPF, therefore disregarding it results in a convex relaxation of the problem:

Model 2.3: The Semidefinite Programming relaxation of the OPF problem

---

variables for each $(i,j) \in E$:

$p_{ij},\ q_{ij}$

variables for each $(i, j) \in N \times N$:

$w^R_{ij} \in [(\boldsymbol{w}^R_{ij})^l,\ (\boldsymbol{w}^R_{ij})^u],\ w^I_{ij} \in [(\boldsymbol{w}^I_{ij})^l,\ (\boldsymbol{w}^I_{ij})^u]$

variables for each $i \in N$:

$w_i \in [(\boldsymbol{v}^l_i)^2,\ (\boldsymbol{v}^u_i)^2]$

$p^g_i \in [\boldsymbol{p}^{\boldsymbol{gl}}_i,\ \boldsymbol{p}^{\boldsymbol{gu}}_i],\ q^g_i \in [\boldsymbol{q}^{\boldsymbol{gl}}_i,\ \boldsymbol{q}^{\boldsymbol{gu}}_i]$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i}(p^g_i)^2 + \boldsymbol{c}_{1i}p^g_i + \boldsymbol{c}_{0i}\right)$$

subject to:

$$p^g_i - \boldsymbol{p}^d_i = \sum_{(i,j)\in E} p_{ij} + \sum_{(j,i)\in E} p_{ij} \ \ \forall i \in N \tag{2.19a}$$

$$q^g_i - \boldsymbol{q}^d_i = \sum_{(i,j)\in E} q_{ij} + \sum_{(j,i)\in E} q_{ij} \ \ \forall i \in N \tag{2.19b}$$

$$p_{ij} = \mathbf{g}_{ij}w_i - \mathbf{g}_{ij}w^R_{ij} - \mathbf{b}_{ij}w^I_{ij} \ \ \forall (i,j) \in E \tag{2.19c}$$

$$q_{ij} = -\mathbf{b}_{ij}w_i + \mathbf{b}_{ij}w^R_{ij} - \mathbf{g}_{ij}w^I_{ij} \ \ \forall (i,j) \in E \tag{2.19d}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{s}^{\boldsymbol{u}}_{ij} \ \ \forall (i,j) \in E \tag{2.19e}$$

$$w^R_{ij}\tan(\boldsymbol{\theta}^l_{ij}) \leqslant w^I_{ij} \leqslant w^R_{ij}\tan(\boldsymbol{\theta}^u_{ij}) \ \ \forall (i,j) \in E \tag{2.19f}$$

$$W \succeq 0 \tag{2.19g}$$

The SDP formulation can be relaxed by disregarding all submatrices of size larger than two. The condition $W \succeq 0$ is replaced with constraints requiring that principal minors of size 2 are nonnegative:

$$(w^R_{ij})^2 + (w^I_{ij})^2 \leqslant w_i w_j \ \forall (i,j) \in E.$$

Since the inequalities for minors of size 1 ($w_i \geqslant 0$) are dominated by squared voltage bounds, they are not included into the model.

This formulation is known as the Second Order Cone Programming (SOCP) relaxation. To the best of our knowledge, it was first proposed by Jabr [102] for radial networks. On such networks the above inequality is strictly equivalent to the full SDP constraint $W \succeq 0$ [173]. In other cases, although the bounds produced by the SOC relaxation are often weak, due to its computational efficiency this model can be used as a basis for dynamic constraint generation approaches [99, 112, 138].

2.4.2 Quadratic Convex relaxation

Quadratic Convex (QC) relaxation was first proposed by Hijazi et al. [100]. As in the SDP relaxation, the w^R_{ij}, w^I_{ij} and w_i variables represent the nonlinear nonconvex terms $v_iv_j\cos(\theta_{ij})$, $v_iv_j\sin(\theta_{ij})$ and v^2_i, and the power flow equations written with the use of these variables become linear. Each nonlinear term is treated as a composition of functions whose convex relaxations are formulated with the use of linear and quadratic constraints and combined in order to obtain the convex relaxation of the whole expression. This approach benefits from the typically small variable bounds in OPF problems, which can be further tightened by applying bound propagation [43].

The QC relaxation has been shown to yield better lower bounds than the SDP relaxation on some instances, which suggests that it captures those aspects of the original problem structure that the SDP formulation fails to account for [100, 43].

To construct the Quadratic Convex formulation, new variables capturing the convex relaxations of the underlying functions are introduced. Let cs_{ij} and sn_{ij} stand respectively for the convex relaxations of the cosine and sine functions of θ_{ij}. w_{ij} will denote the relaxation of the bilinear product $v_i v_j$. Consistently with the notation used for describing the SDP relaxation, variables w_i, w^R_{ij} and w^I_{ij} represent the expressions v_i^2, $v_i v_j \cos(\theta_{ij})$ and $v_i v_j \sin(\theta_{ij})$ respectively. The following bounds on the new variables are enforced:

$$\cos(\boldsymbol{\theta}^u_{ij}) \leqslant cs_{ij} \leqslant 1, \tag{2.20}$$

$$-\sin(\boldsymbol{\theta}^u_{ij}) \leqslant sn_{ij} \leqslant \sin(\boldsymbol{\theta}^u_{ij}), \tag{2.21}$$

$$\boldsymbol{v}^l_i \boldsymbol{v}^l_j \leqslant w_{ij} \leqslant \boldsymbol{v}^u_i \boldsymbol{v}^u_j, \tag{2.22}$$

$$(\boldsymbol{v}^l_i)^2 \leqslant w_i \leqslant (\boldsymbol{v}^u_i)^2, \tag{2.23}$$

$$\boldsymbol{v}^l_i \boldsymbol{v}^l_j \cos(\boldsymbol{\theta}^u_{ij}) \leqslant w^R_{ij} \leqslant \boldsymbol{v}^u_i \boldsymbol{v}^u_j, \tag{2.24}$$

$$-\boldsymbol{v}^l_i \boldsymbol{v}^l_j \sin(\boldsymbol{\theta}^u_{ij}) \leqslant w^I_{ij} \leqslant \boldsymbol{v}^u_i \boldsymbol{v}^u_j \sin(\boldsymbol{\theta}^u). \tag{2.25}$$

Trigonometric functions It is assumed that $-\boldsymbol{\theta}^l = \boldsymbol{\theta}^u \leqslant \pi/2$. Hijazi et al. [100] have proven that the following functions over- or underestimate trigonometric functions on $[-\boldsymbol{\theta}^u, \boldsymbol{\theta}^u]$:

Proposition 2.1. *[100] Let*

$$\widehat{cs}(\theta) = 1 - \frac{1 - \cos(\boldsymbol{\theta}^u)}{(\boldsymbol{\theta}^u)^2}\theta^2.$$

Then $\widehat{cs}(\theta) \geqslant \cos(\theta)\ \forall \theta \in [-\boldsymbol{\theta}^u, \boldsymbol{\theta}^u]$.

The above proposition provides a quadratic overestimator of $\cos(\theta)$. Since for $\theta \in [-\boldsymbol{\theta}^u, \boldsymbol{\theta}^u]$ the cosine function decreases as $|\theta|$ increases, it is easy to see that $\cos(\boldsymbol{\theta}^u) \leqslant \cos(\theta)\ \forall \theta \in [-\boldsymbol{\theta}^u, \boldsymbol{\theta}^u]$, therefore $\cos(\boldsymbol{\theta}^u_{ij})$ can be used as the lower bound on cs_{ij}.

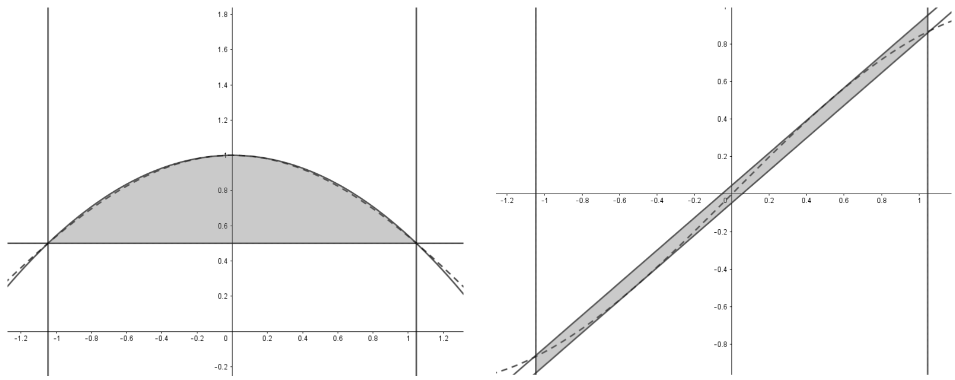
Proposition 2.2. *[100] Let*

$$\widehat{sn}(\theta) = \cos\left(\frac{\boldsymbol{\theta}^u}{2}\right)\left(\theta - \frac{\boldsymbol{\theta}^u}{2}\right) + \sin\left(\frac{\boldsymbol{\theta}^u}{2}\right),$$

$$\widetilde{sn}(\theta) = \cos\left(\frac{\boldsymbol{\theta}^u}{2}\right)\left(\theta + \frac{\boldsymbol{\theta}^u}{2}\right) - \sin\left(\frac{\boldsymbol{\theta}^u}{2}\right).$$

Then $\breve{sn} \leqslant sin(\theta) \leqslant \widehat{sn}(\theta)\ \forall\theta \in [-\boldsymbol{\theta}^u, \boldsymbol{\theta}^u]$.

These results are used to construct valid convex relaxations of the cosine and sine functions shown in Figure 2.2.



Cosine relaxation　　　　Sine relaxation

Figure 2.2: Convex relaxations of trigonometric terms

Quadratic terms The convex envelope for a quadratic function is written as:

$$w \geqslant v^2,$$
$$w \leqslant (\boldsymbol{v}^l + \boldsymbol{v}^u)v - \boldsymbol{v}^u\boldsymbol{v}^l.$$

Multilinear terms The McCormick formulation [136] is applied to obtain convex relaxations of the bilinear products. For example, given two variables v_i, v_j and their lower and upper bounds, the McCormick relaxation of the product $v_i v_j$ has the form:

$$\begin{aligned}
w_{ij} &\geqslant \boldsymbol{v}_i^l v_j + \boldsymbol{v}_j^l v_i - \boldsymbol{v}_i^l \boldsymbol{v}_j^l, \\
w_{ij} &\geqslant \boldsymbol{v}_i^u v_j + \boldsymbol{v}_j^u v_i - \boldsymbol{v}_i^u \boldsymbol{v}_j^u, \\
w_{ij} &\leqslant \boldsymbol{v}_i^l v_j + \boldsymbol{v}_j^u v_i - \boldsymbol{v}_i^l \boldsymbol{v}_j^u, \\
w_{ij} &\leqslant \boldsymbol{v}_i^u v_j + \boldsymbol{v}_j^l v_i - \boldsymbol{v}_i^u \boldsymbol{v}_j^l.
\end{aligned}$$

The set of numbers w_{ij} satisfying the above inequalities will be denoted as $MC(v_i, v_j)$. The relaxations of the trilinear terms are formulated by applying sequential McCormick relaxations. Thus the relaxation of $v_i v_j \cos(\theta_{ij})$ is written as $MC(w_{ij}, cs_{ij})$ and the relaxation of $v_i v_j \sin(\theta_{ij})$ is written as $MC(w_{ij}, sn_{ij})$.

Using trilinear product relaxations instead of McCormick formulations does not lead to an improvement in computational results [100].

Strengthening the model The QC relaxation can be strengthened by introducing inequalities that are derived by considering line power losses:

$$p_{ij} + p_{ji} = \boldsymbol{r}_{ij} l_{ij}, \tag{2.26}$$

$$q_{ij} + q_{ji} = \boldsymbol{x}_{ij} l_{ij}, \tag{2.27}$$

$$l_{ij} \geqslant \frac{p_{ij}^2 + q_{ij}^2}{w_i}, \tag{2.28}$$

and the bounds on the phase angle differences expressed in terms of the w_{ij}^R, w_{ij}^I variables:

$$\tan(\boldsymbol{\theta}_{ij}^{\boldsymbol{l}}) w_{ij}^R \leqslant w_{ij}^I \leqslant \tan(\boldsymbol{\theta}_{ij}^{\boldsymbol{u}}) w_{ij}^R.$$

The Quadratic Convex formulation The full formulation of the Quadratic Convex relaxation of OPF is given by Model 2.4.

Model 2.4: The Quadratic Convex relaxation of the OPF problem

---

variables for each $(i, j) \in E$:

$p_{ij},\ q_{ij}$

$cs_{ij} \in [\cos(\boldsymbol{\theta}_{ij}^u),\ 1],\ sn_{ij} \in [-\sin(\boldsymbol{\theta}_{ij}^u),\ \sin(\boldsymbol{\theta}_{ij}^u)]$

$w_{ij} \in [\boldsymbol{v}_i^l \boldsymbol{v}_j^l,\ \boldsymbol{v}_i^u \boldsymbol{v}_j^u],\ w_{ij}^R \in [\boldsymbol{v}_i^l \boldsymbol{v}_j^l \cos(\boldsymbol{\theta}_{ij}^u),\ \boldsymbol{v}_i^u \boldsymbol{v}_j^u]$

$w_{ij}^I \in [-\boldsymbol{v}_i^l \boldsymbol{v}_j^l \sin(\boldsymbol{\theta}_{ij}^u),\ \boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}^u)]$

$\theta_{ij} \in [-\boldsymbol{\theta}_{ij}^u,\ \boldsymbol{\theta}_{ij}^u]$

variables for each $i \in N$:

$w_i \in [(\boldsymbol{v}_i^l)^2,\ (\boldsymbol{v}_i^u)^2]$

$p_i^g \in [\boldsymbol{p}_i^{\boldsymbol{gl}},\ \boldsymbol{p}_i^{\boldsymbol{gu}}],\ q_i^g \in [\boldsymbol{q}_i^{\boldsymbol{gl}},\ \boldsymbol{q}_i^{\boldsymbol{gu}}]$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i} (p_i^g)^2 + \boldsymbol{c}_{1i} p_i^g + \boldsymbol{c}_{0i} \right)$$

subject to:

$$\theta_{\boldsymbol{r}} = 0 \tag{2.29a}$$

$$p_i^g - \boldsymbol{p}_i^d = \sum_{(i,j) \in E} p_{ij} + \sum_{(j,i) \in E} p_{ij} \quad \forall i \in N \tag{2.29b}$$

$$q_i^g - \boldsymbol{q}_i^d = \sum_{(i,j) \in E} q_{ij} + \sum_{(j,i) \in E} q_{ij} \quad \forall i \in N \tag{2.29c}$$

$$p_{ij} = \mathbf{g}_{ij} w_i - \mathbf{g}_{ij} w_{ij}^R - \mathbf{b}_{ij} w_{ij}^I \quad \forall (i, j) \in E \tag{2.29d}$$

$$q_{ij} = -\mathbf{b}_{ij} w_i + \mathbf{b}_{ij} w_{ij}^R - \mathbf{g}_{ij} w_{ij}^I \quad \forall (i, j) \in E \tag{2.29e}$$

$$-\tan(\boldsymbol{\theta}_{ij}^{\boldsymbol{u}}) w_{ij}^R \leqslant w_{ij}^I \leqslant \tan(\boldsymbol{\theta}_{ij}^{\boldsymbol{u}}) w_{ij}^R \quad \forall (i, j) \in E \tag{2.29f}$$

$$\cos(\boldsymbol{\theta}^u_{ij}) \leqslant cs_{ij} \leqslant \widehat{cs}(\theta_i - \theta_j) \quad \forall (i,j) \in E \tag{2.29g}$$

$$\widetilde{sn}(\theta_i - \theta_j) \leqslant sn_{ij} \leqslant \widehat{sn}(\theta_i - \theta_j) \quad \forall (i,j) \in E \tag{2.29h}$$

$$v_i^2 \leqslant w_i \leqslant (\boldsymbol{v}^l + \boldsymbol{v}^u)v - \boldsymbol{v}^u\boldsymbol{v}^l \quad \forall i \in N \tag{2.29i}$$

$$w_{ij} \in MC(v_i, v_j) \quad \forall (i,j) \in E \tag{2.29j}$$

$$w^R_{ij} \in MC(w_{ij}, cs_{ij}) \quad \forall (i,j) \in E \tag{2.29k}$$

$$w^I_{ij} \in MC(w_{ij}, sn_{ij}) \quad \forall (i,j) \in E \tag{2.29l}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{s}^{\boldsymbol{u}}_{ij} \quad \forall (i,j) \in E \tag{2.29m}$$

$$(2.26) - (2.28) \tag{2.29n}$$

2.4.3 Linear relaxations

Linear relaxations generally lack the lower bound strength of the nonlinear formulations but have the advantage of fast performance. Moreover, when added to the model in a dynamic iterative process, linear constraints can produce tight bounds. Bienstock and Muñoz [22] proposed a linear formulation in a lifted space, deriving cuts that can be used to strengthen relaxations of OPF iteratively. The paper by Taylor and Hover [181] presents a relaxation based on network flow models. Coffrin et al. [41] continued this line of work and developed two relaxations. One is close to that introduced by Taylor and Hover [181], but enforces nonnegative line power losses (since power cannot be generated on lines). The other derived in a similar fashion to the copper plate approximations which are based on the balance of supply and demand throughout the network.

2.5 Optimal Transmission Switching

Line switching equipment allows to change the configuration of the network dynamically and thus enables significant savings [59, 93, 94, 71, 158]. Topology design for reducing generation costs was originally suggested by O'Neill et al. [152] and formalised by Fisher et al. [58], and is referred to as Optimal Transmissions Switching (OTS).

From a mathematical standpoint, OTS is a mixed-integer nonlinear nonconvex problem. A binary variable z_{ij} is associated with each line $(i,j) \in E$ and represents its state. If $z_{ij} = 0$ (the line is switched off), then the power flow has to be set to zero since a deactivated line can transmit no power. This is achieved by modifying the Kirchhoff's current law and thermal limit constraints. The phase angle differences are affected by line switching: their lower and upper bounds are replaced by constants $\boldsymbol{M}^l$ and $\boldsymbol{M}^u$ using the same reasoning as explained in Subsection 2.4.1.

Model 2.5 shows the OTS problem in the polar form. Here the notation

$$x_{ij} \in [\boldsymbol{x}^{l1}_{ij}, \boldsymbol{x}^{u1}_{ij}; \boldsymbol{x}^{l0}_{ij}, \boldsymbol{x}^{u0}_{ij}]$$

is used to specify on/off bounds on a variable and is equivalent to

$$\begin{cases} \boldsymbol{x}_{ij}^{l1} \leqslant x_{ij} \leqslant \boldsymbol{x}_{ij}^{u1} \text{ if } z_{ij} = 1, \\ \boldsymbol{x}_{ij}^{l0} \leqslant x_{ij} \leqslant \boldsymbol{x}_{ij}^{l0} \text{ if } z_{ij} = 0. \end{cases}$$

Model 2.5: The Optimal Transmission Switching problem, polar form

variables for each $(i, j) \in E$:

$p_{ij},\ q_{ij}\ z_{ij} \in \{0, 1\}$

$\theta_{ij} \in [-\boldsymbol{\theta}_{ij}^{u},\ \boldsymbol{\theta}_{ij}^{u};\ \ -\boldsymbol{M}^{u},\ \boldsymbol{M}^{u}]$

variables for each $i \in N$:

$v_i \in [\boldsymbol{v}_i^{l},\ \boldsymbol{v}_i^{u}],\ \theta_i$

$p_i^g \in [\boldsymbol{p}_i^{gl},\ \boldsymbol{p}_i^{gu}],\ q_i^g \in [\boldsymbol{q}_i^{gl},\ \boldsymbol{q}_i^{gu}]$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i}(p_i^g)^2 + \boldsymbol{c}_{1i} p_i^g + \boldsymbol{c}_{0i}\right) \tag{2.30a}$$

subject to:

$$\theta_{\boldsymbol{r}} = 0 \tag{2.30b}$$

$$\theta_{ij} = \theta_i - \theta_j \quad \forall (i, j) \in E \tag{2.30c}$$

$$p_i^g - \boldsymbol{p}_i^d = \sum_{(i,j) \in E} p_{ij} z_{ij} + \sum_{(j,i) \in E} p_{ij} z_{ij} \quad \forall i \in N \tag{2.30d}$$

$$q_i^g - \boldsymbol{q}_i^d = \sum_{(i,j) \in E} q_{ij} z_{ij} + \sum_{(j,i) \in E} q_{ij} z_{ij} \quad \forall i \in N \tag{2.30e}$$

$$p_{ij} = \mathbf{g}_{ij} v_i^2 - \mathbf{g}_{ij} v_i v_j \cos(\theta_{ij}) - \mathbf{b}_{ij} v_i v_j \sin(\theta_{ij}) \quad \forall (i, j) \in E \tag{2.30f}$$

$$q_{ij} = -\mathbf{b}_{ij} v_i^2 + \mathbf{b}_{ij} v_i v_j \cos(\theta_{ij}) - \mathbf{g}_{ij} v_i v_j \sin(\theta_{ij}) \quad \forall (i, j) \in E \tag{2.30g}$$

$$p_{ij}^2 + q_{ij}^2 \leqslant \boldsymbol{s}_{ij}^{\boldsymbol{u}} z_{ij} \quad \forall (i, j) \in E \tag{2.30h}$$

2.6 Relaxations of OTS

In addition to the same non-convexities as in the OPF problem, OTS contains binary variables, which makes it even more challenging.

To solve the OTS problem, many studies [58, 93, 94, 70, 14, 15, 92, 16] apply DC approximations. However, recent research [40] has shown that the latter does not appear to be appropriate for OTS studies as it exhibits significant feasibility issues with respect to the original nonlinear model. Moreover, the approximate linear formulation can either underestimate or overestimate the benefits of line switching in different contexts. This motivates the use of convex relaxations in the case of OTS.

2.6.1 Quadratic Convex relaxation

To the best of our knowledge, the first convex relaxation proposed for the OTS problem is the QC relaxation [100].

Since OTS is an extension of OPF, the QC relaxation of the latter can be modified in order be applied to the former. Therefore we will not restate this formulation but will only describe the changes.

The main feature of OTS is that the power flows p_{ij}, q_{ij} along a line become zero when line (i, j) is switched off (i.e. when $z_{ij} = 0$). The modifications of the QC model must ensure that this condition is satisfied and preserve convexity of the continuous relaxation, since it is necessary so that the convex MINLP solvers would provide global optimality guarantees. Moreover, formulations that result in tight continuous relaxations are desirable since they tend to improve performance.

In order to achieve this, disjunctive constraints are introduced that affect the feasible region only when the corresponding line is activated. To express these constraints algebraically, a well known method referred to as big-M relaxation [143] is used unless stated otherwise. For a general on/off constraint of the form $g(\mathbf{x}) \leqslant 0$ if $z = 1$, the big-M formulation is written as $g(\mathbf{x}) \leqslant (1 - z)\boldsymbol{M}$, where $\boldsymbol{M}$ is a constant that should be large enough so that the constraint becomes redundant at $z = 0$ (hence the name "big-M").

To ensure that variables w^R_{ij}, w^I_{ij}, cs_{ij}, sn_{ij} have zero values whenever the corresponding line is deactivated, on/off variable bounds are added to the model.

Cosine disjunction Given the upper bound on θ denoted as $\boldsymbol{\theta^u}$ and the cardinality of the set of arcs $|E|$, let $CS^{0\text{-}1}$ denote the set of (cs, θ, z) that satisfy the quadratic relaxation of the on/off constraint: $cs = \cos(\theta)$ if $z = 1$. Then $(cs, \theta, z) \in CS^{0\text{-}1}$ if:

$$cs \leqslant z - \frac{1 - \cos(\boldsymbol{\theta}^u)}{(\boldsymbol{\theta}^u)^2}\theta^2 + (1 - z)\frac{1 - \cos(\boldsymbol{\theta}^u)}{(\boldsymbol{\theta}^u)^2}(|E|(\boldsymbol{\theta}^u)^2).$$

Using this definition, the relaxation of cosine for each line $(i, j) \in E$ can be written as:

$$(cs_{ij}, \theta_{ij}, z_{ij}) \in CS^{0\text{-}1}_{ij}.$$

Sine disjunction For the formulation of the on/off version of the sine relaxation, the results from [101] are used in the QC-OTS model [100]. For the disjunctive version of a linear constraint of the form $\boldsymbol{a}^T\mathbf{x} - \boldsymbol{b} \leqslant 0$, where $\mathbf{x} = (x_1, \ldots, x_n)^T$, a big-M-like algebraic formulation is given by:

$$\sum_{i=1}^{n} \boldsymbol{a}_i x_i \leqslant \boldsymbol{b}z + (1 - z)\left(\sum_{\substack{i=1\\ \boldsymbol{a}_i<0}}^{n} \boldsymbol{a}_i \boldsymbol{x}^{l0} + \sum_{\substack{i=1\\ \boldsymbol{a}_i>0}}^{n} \boldsymbol{a}_i \boldsymbol{x}^{u0}\right). \tag{2.31}$$

Let $SN^{0\text{-}1}$ denote the set of (sn, θ, z) that satisfy the sine relaxation. Using formulation

(2.31), we express it by the following inequalities:

$$sn - \cos(\boldsymbol{\theta^u_{1/2}})\theta \leqslant z(\sin(\boldsymbol{\theta^u_{1/2}}) - \cos(\boldsymbol{\theta^u_{1/2}})\boldsymbol{\theta^u_{1/2}}) + (1-z)(\cos(\boldsymbol{\theta^u_{1/2}})|E|\boldsymbol{\theta}^u + 1),$$
$$\cos(\boldsymbol{\theta^u_{1/2}})\theta - sn \leqslant z(\sin(\boldsymbol{\theta^u_{1/2}}) - \cos(\boldsymbol{\theta^u_{1/2}})\boldsymbol{\theta^u_{1/2}}) + (1-z)(\cos(\boldsymbol{\theta^u_{1/2}})|E|\boldsymbol{\theta}^u + 1),$$

where $\boldsymbol{\theta^u_{1/2}} = \boldsymbol{\theta}^u/2$.

The relaxation will require that for each line $(i,j) \in E$ the following holds:

$$(sn_{ij}, \theta_{ij}, z_{ij}) \in SN^{0\text{-}1}_{ij}.$$

Current magnitude disjunction The current magnitude disjunction is expressed by a set of quadratic and linear constraints:

$$\begin{aligned}
&p_{ij}^2 + q_{ij}^2 \leqslant (\boldsymbol{v}^u)^2 l_{ij} z_{ij} && \forall (i,j) \in E \\
&p_{ij}^2 + q_{ij}^2 \leqslant \boldsymbol{l}^u_{ij} w_i z_{ij} && \forall (i,j) \in E \\
&l_{ij} = (\boldsymbol{g}_{ij}^2 + \boldsymbol{b}_{ij}^2)(w_i + w_j - 2w^R_{ij}) && \forall (i,j) \in E
\end{aligned}$$

Variable bounds The bounds for each variable where they differ in the 'off' and 'on' states are expressed using a general formula:

$$\boldsymbol{x}^{l_1} z + \boldsymbol{x}^{l_0}(1-z) \leqslant x \leqslant \boldsymbol{x}^{u_1} z + \boldsymbol{x}^{u_0}(1-z),$$

where $\boldsymbol{x}^{l_1}$, $\boldsymbol{x}^{u_1}$ represent the lower and upper bounds when the corresponding binary variable is equal to one and $\boldsymbol{x}^{l_0}$, $\boldsymbol{x}^{u_0}$ denote the bounds when the binary is equal to zero.

Finally, McCormick relaxations (2.29j)-(2.29l) are rewritten with the use of a big-M formulation and thermal limit constraints are identical to constraints (2.30h) in the nonconvex OTS problem.

2.6.2 MISOCP relaxation

An approach based on the second order cone formulation has been proposed by Kocuk et al. [113]. In order to linearise the power flow equations, it uses the same variables w_i, w^R_{ij} and w^I_{ij} as the previously described formulations and introduces a new set of variables w^j_i which represent the on/off squared voltage at node i which is set to zero when line (i,j) is deactivated. These variables are characterised by the following constraints:

$$(\boldsymbol{w}^R_{ij})^l z_{ij} \leqslant w^R_{ij} \leqslant (\boldsymbol{w}^R_{ij})^u z_{ij}, \tag{2.32}$$
$$(\boldsymbol{w}^I_{ij})^l z_{ij} \leqslant w^I_{ij} \leqslant (\boldsymbol{w}^I_{ij})^u z_{ij}, \tag{2.33}$$
$$(\boldsymbol{v}^l_i)^2 z_{ij} \leqslant w^j_i \leqslant (\boldsymbol{v}^u_i)^2 z_{ij}, \tag{2.34}$$

$$w_i - (\boldsymbol{v}_i^u)^2(1 - z_{ij}) \leqslant w_i^j, \tag{2.35}$$

$$w_i^j \leqslant w_i - (\boldsymbol{v}_i^l)^2(1 - z_{ij}), \tag{2.36}$$

$$(w_{ij}^R)^2 + (w_{ij}^I)^2 \leqslant w_i^j w_j^i, \tag{2.37}$$

where (2.32) and (2.33) represent the on/off bounds on w_{ij}^R and w_{ij}^I, constraints (2.34)-(2.36) are obtained by applying the McCormick relaxation to $w_i^j = w_i z_{ij}$ and inequality (2.37) is the second order cone constraint.

To strengthen the formulation, additional inequalities are applied as cuts based on constraints that connect the phase angle difference with variables w_{ij}^R and w_{ij}^I:

$$(\theta_j - \theta_i - \text{atan2}(w_{ij}^I, w_{ij}^R))z_{ij} = 0,$$

as well as SDP conditions and a cycle-based OPF formulation that was proposed by the same authors [112].

2.7 Benchmarks

For the computational experiments we use version 17.08 of the Power Grid Lib (PGlib) benchmarks which can be found at `https://github.com/power-grid-lib/pglib-opf`.

The benchmark contains three sets of instances: standard instances with typical operating conditions and more challenging Active Power Increase (API) and Small Angle Difference (SAD) instances with network sizes ranging from 3 to 9241 nodes.

The API instances represent networks with congested operating conditions where the line thermal limits are binding. It has been observed that optimal network topology depends on the loading scenarios [59], and considering networks where power flow congestion occurs leads to interesting OTS cases. The API instances are obtained by removing bounds on generator power output and solving active power increase problems which increase active loads until line thermal capacities become binding. The values of power generation and loads then define the new test cases [39].

Small phase angle differences provide better power system stability and impact optimisation approaches [44, 45, 100]. The SAD instances are constructed by solving small angle difference problems, where the objective is to find the minimum phase angle difference bound that, when applied to all lines in the network, does not result in infeasibility. Reduced angle differences provide guarantees in terms of power systems stability and impact optimization approaches, resulting in more challenging test cases with larger optimality gaps.

Both scenarios increase optimality gaps, leading to more challenging test cases [39].

Chapter 3

Conditions for KT-Invexity

3.1 Introduction

In this chapter we focus on a special class of generalised convex problems called KT-invex problems, for which every point satisfying KKT conditions (2.1)-(2.5) is the global optimum. The polynomial time algorithms that converge to a local optimum in the general case will always find the global optimum for invex problems. This is often the case for OPF problems with realistic parameters. However, KT-invexity needs to be proven theoretically in order to guarantee global optimality for such problems.

The main idea behind generalised convexity is to identify the key features of convex functions and problems from a global optimisation point of view. As we will discuss in more detail below, the most popular approach has been to start with the well known characterisation of convex functions: given a convex set $C \in \mathbb{R}^n$ and a function $f: \mathbb{R}^n \to \mathbb{R}$,

$$f \text{ is convex on } C \;\Leftrightarrow\; f(\mathbf{x}) - f(\mathbf{y}) \geqslant (\mathbf{x}-\mathbf{y})^T \cdot \nabla f(\mathbf{y}) \;\forall\; \mathbf{x}, \mathbf{y} \in C, \tag{3.1}$$

and propose different relaxations of this condition.

This work introduces a new way of looking at KT-invexity of optimisation problems. We notice that for maximisation problems with a concave objective, the properties of the boundary of the feasible set define whether the problem is KT-invex. More specifically, it is the behaviour of the objective function on the boundary that is key to determining KT-invexity. There exists a subset of "problematic" points whose existence leads to multiple local optima. These points can be found by studying different sections of the boundary corresponding to individual constraints.

This chapter defines a property, which we refer to as boundary-invexity, that is requiring that a problem does not have any "problematic" points. We prove its necessity for KT-invexity in the case of n-dimensional problems and establish the equivalence between boundary-invexity and KT-invexity for problems with two degrees of freedom.

This chapter is organised as follows.

Section 3.2 looks into the development of generalised convexity notions in the literature and restates some key results on KT-invexity and its variations. In Section 3.3 we introduce the notion of boundary-invexity, prove its necessity for KT-invexity and study its connection to the local optimality of KKT points. Here we also establish the connection between global optimality on the boundary and in the interior.

After proving these results, we proceed to laying the technical foundation for the more challenging proof of sufficiency of boundary-invexity for KT-invexity of problems with two degrees of freedom. Section 3.4 proves some properties of a pseudo-scalar product of vectors. In Section 3.5 we define a parametrisation of the boundary curve. In Section 3.6 we study the behaviour of concave functions on a line and present some results on boundary-optimality.

Section 3.7 presents the main theorem establishing the sufficiency of boundary-invexity for two-dimensional problems. Boundary-invexity of the Optimal Power Flow problem is investigated in Section 3.8. Finally, Section 3.9 concludes the chapter and discusses ideas for future work.

3.2 Generalised Convexity

3.2.1 Early generalisations

Many practical applications involve non-convexities, and therefore the results proven for convex problems cannot be directly applied to them. However, convexity is not necessary for some convenient properties of optimisation problems to hold.

This has motivated research on finding various relaxations of convexity that preserve certain key characteristics of convex problems. Among the first generalised convexity concepts were pseudo- and quasi-convexity proposed by Mangasarian [131]. These definitions were obtained by relaxing condition (3.1) by replacing the inequality with an implication. One such relaxation leads to the definition of pseudo-convex functions:

$$f \text{ is pseudo-convex on } C \;\Leftrightarrow$$
$$(\mathbf{x}-\mathbf{y})^T \cdot \nabla f(\mathbf{y}) \geqslant 0 \;\Rightarrow\; f(\mathbf{x}) \geqslant f(\mathbf{y}) \;\forall\; \mathbf{x},\mathbf{y} \in C,$$

and another results in the definition of quasi-convex functions:

$$f \text{ is quasi-convex on } C \;\Leftrightarrow$$
$$f(\mathbf{x}) \leqslant f(\mathbf{y}) \;\Rightarrow (\mathbf{x}-\mathbf{y})^T \cdot \nabla f(\mathbf{y}) \leqslant 0 \;\;\forall\; \mathbf{x},\mathbf{y} \in C.$$

Mangasarian has shown [131] that every local minimum of a pseudo-convex function is its global minimum. In the same work it was proved that every KKT point of a minimisation problem with quasi-convex constraints of the form $g_i(\mathbf{x}) \leqslant 0$ and a pseudo-convex objective is a global optimiser.

3.2.2 Invexity

It can be noticed that the main proofs of the properties of problems involving convex and generalised convex functions do not explicitly depend on the linear term in the definition, thus allowing for further relaxations. Therefore another direction of generalisation is based on modifying the expressions in the definition of convexity (3.1) ([5, 6, 19, 33]). In invex functions introduced by Hanson [91] the linear term is replaced by an arbitrary vector function η:

Definition 3.1. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is called invex on C if*

$$f(\mathbf{x}) - f(\mathbf{y}) \geqslant \eta(\mathbf{x}, \mathbf{y}) \cdot \nabla f(\mathbf{y}) \ \forall \mathbf{x}, \mathbf{y} \in C$$

for some arbitrary vector function $\eta : C \times C \to \mathbb{R}^n$.

The term "invex" was introduced by Craven [49], meaning "invariant convex", since an invex function can be created by taking a composition of a differential injective coordinate transformation and a convex function. Ben-Israel and Mond [18] proved that a function is invex if and only if its every stationary point is a global minimum.

In constrained optimisation, invexity of functions has a connection to a property known as KT-invexity. Consider the optimisation problem:

$$\begin{aligned} \max\ & f(\mathbf{x}) \\ \text{s.t. } & g_i(\mathbf{x}) \leqslant 0 \ \forall i = 1, \ldots, m, \\ & \mathbf{x} \in \mathbb{R}^n, \end{aligned} \tag{NLP}$$

where functions f and g_i $(i = 1, \ldots, m)$ are twice continuously differentiable and f is concave. The results in this chapter can be extended to problems with pseudoconcave objective functions since only convexity of the superlevel sets of f and the signs of $f(\mathbf{x}) - f(\mathbf{y})$ and $(\nabla f(\mathbf{y}))^T \cdot (\mathbf{x} - \mathbf{y})$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ are used in the proofs. Let F denote the feasible set of (NLP).

Definition 3.2. *[133] An optimisation problem is said to be Kuhn-Tucker invex (KT-invex) if every KKT point is a global optimiser.*

Hanson [91] proved that KT-invexity holds if all functions in a problem are invex with respect to the same vector function η. Craven [47] proposed an equivalent characterisation of the latter property by requiring that a vector function that depends on all functions in the problem and some vector function η is nonnegative. In the same work, based on this characterisation, K-invex vector functions were defined by relaxing the non-negativity condition and requiring that the values of the vector function belong to a cone. Another characterisation was introduced by Martin [133] by relaxing the invexity condition on the constraints and enforcing it only on the boundary of the feasible set. This condition has been proved to be necessary and sufficient for KT-invexity.

Theorem 3.1. *[133] (NLP) is KT-invex if and only if there exists a function $\eta : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ such that*

$$\mathbf{x}, \mathbf{y} \in F \Rightarrow$$

$$\begin{cases} f(\mathbf{x}) - f(\mathbf{y}) - \eta(\mathbf{x}, \mathbf{y}) \cdot \nabla f(\mathbf{y}) \geqslant 0 \\ g_i(\mathbf{y}) = 0 \Rightarrow \eta(\mathbf{x}, \mathbf{y}) \cdot \nabla g_i(\mathbf{y}) \geqslant 0 \ \forall i = 1, \ldots, m. \end{cases}$$

Practical applicability of this condition is limited by the need to find an η that will satisfy the inequalities for the objective function as well as all constraints.

Alternative formulations of the invexity conditions have been suggested in an attempt to tackle this difficulty. A characterisation that does not require finding a common function η was proposed by Craven [48]. Consider a twice differentiable vector function $F: \mathbb{R}^n \to \mathbb{R}^m$. Observe that a vector function being invex is equivalent to all its components being invex with respect to the same η, and these properties can be used interchangeably. Craven gives the following condition for the invexity of F:

Theorem 3.2. *[48] F is invex at $\mathbf{x}^*$ if and only if*

$$(0 \neq \alpha \in \mathbb{R}^m_+, \ \alpha^T \cdot \nabla F(\mathbf{x}^*) = 0) \Rightarrow \alpha^T \cdot F^{\#}(\mathbf{x} - \mathbf{x}^*, \mathbf{x}^*) \geqslant 0,$$

where $F^{\#}(\mathbf{x} - \mathbf{x}^*, \mathbf{x}^*)$ represents the higher-order terms in the Taylor expansion of F:

$$F^{\#}(\mathbf{x} - \mathbf{x}^*, \mathbf{x}^*) = F(\mathbf{x}) - F(\mathbf{x}^*) - (\nabla F(\mathbf{x}^*))^T \cdot (\mathbf{x} - \mathbf{x}^*).$$

However, the evaluation of $F^{\#}(\mathbf{x} - \mathbf{x}^*, \mathbf{x}^*)$ is not straightforward. Another characterisation has been derived [48] based on Theorem 3.2:

Theorem 3.3. *[48] Consider the Wolfe dual for (NLP):*

$$\max \left(-f(\mathbf{u}) + \mathbf{v}^T \cdot G(\mathbf{u})\right) \ \text{s.t.} \ \mathbf{v} \geqslant 0, \ -\nabla f(\mathbf{u}) + \mathbf{v}^T \cdot \nabla G(\mathbf{u}) = 0,$$

where $G: \mathbb{R}^n \to \mathbb{R}^m$ is a vector comprised of all constraint functions g_i.

For each feasible point $(\mathbf{u}, \mathbf{v})$, assume that $\nabla G(\mathbf{u})$ has full rank, $\nabla f(\mathbf{u}) \neq 0$, and

$$-f(\mathbf{z}) + \mathbf{v}^T \cdot G(\mathbf{z}) \geqslant -f(\mathbf{u}) + \mathbf{v}^T \cdot G(\mathbf{u}) \ \forall \mathbf{z}. \tag{3.2}$$

Then (f, G) is invex at $(\mathbf{u}, \mathbf{v})$.

This condition is difficult to verify because (3.2) needs to be checked for every feasible point $(\mathbf{u}, \mathbf{v})$. Moreover, identifying whether $\mathbf{u}$ is the global minimum of $-f(\mathbf{z}) + \mathbf{v}^T \cdot G(\mathbf{z})$ over all $\mathbf{z}$ is NP-hard since it is a nonconvex problem.

A different approach has been introduced by Martínez-Legaz [135] by considering linear combinations of functions:

Theorem 3.4. *[135] A differentiable vector function $F: \mathbb{R}^n \to \mathbb{R}^m$ is invex if and only if the function $\sum_{i=1}^{m} \lambda_i f_i$ is invex for all $(\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^m$, where functions $f_i: \mathbb{R}^n \to \mathbb{R}$ for $i = 1, \ldots, m$ represent vector components of F.*

However, checking the invexity of all linear combinations is a difficult problem.

To the best of our knowledge, the lack of algorithmically verifiable conditions still remains a major limitation of the invexity theory which we are starting to address in this chapter.

3.3 New Conditions for Kuhn-Tucker Invexity

Let us emphasise that checking local optimality is NP-hard in general:

Theorem 3.5. *[155] The problem of checking local optimality for a feasible solution of (NLP) is NP-hard.*

In this work, we try to investigate necessary and sufficient conditions that allow us to circumvent the negative result presented in Theorem 3.5 by identifying problems where KKT points are provably global optimisers.

3.3.1 Weak boundary-invexity

For each nonconvex constraint $g_i(x) \leqslant 0$ define the problem:

$$\begin{aligned} &\min \;\; f(\mathbf{x}) \\ &s.t. \; g_i(\mathbf{x}) = 0. \end{aligned} \tag{NLP$_i$}$$

The objective in (NLP$_i$) is opposite to that in (NLP), for example, if a function is to be maximised in (NLP), then (NLP$_i$) should minimise the same function.

First let us restate the definition of a strict local minimiser:

Definition 3.3. *[150] A point $\mathbf{x}^* \in \mathbb{R}^n$ is a strict local minimiser for (NLP$_i$) if $g_i(\mathbf{x}^*) = 0$ and there is a neighbourhood $N(\mathbf{x}^*)$ such that $f(\mathbf{x}) > f(\mathbf{x}^*)$ for $\mathbf{x} \in N(\mathbf{x}^*) \setminus \mathbf{x}^* \mid g_i(\mathbf{x}) = 0$.*

Now we can define a new property that we refer to as weak boundary-invexity:

Definition 3.4. *(Weak boundary-invexity) Problem (NLP) is weakly boundary-invex if for every i that corresponds to a nonconvex constraint either the problem (NLP$_i$) does not have a finite global optimal solution $\mathbf{x}^*$ or at least one of the following holds:*

1. *$\mathbf{x}^*$ is infeasible for (NLP),*
2. *$\mathbf{x}^*$ is not a strict minimiser for (NLP$_i$),*
3. *the Lagrange multiplier for $\mathbf{x}^*$ in (NLP$_i$) is nonnegative,*
4. *there exist constraints $g_j(\mathbf{x}) \leqslant 0,\; j \neq i$ in (NLP) that are active at $\mathbf{x}^*$.*

(NLP$_i$) is still a nonconvex problem, and finding its global optimum can be NP-hard in general. However, in some special cases (NLP$_i$) can be more tractable than (NLP) since we are restricting the feasible region to part of its boundary.

For instance, when both f and g_i are quadratic functions we can apply an extension of the S-lemma:

Theorem 3.6. *[193] Let $f(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + a^T \cdot \mathbf{x} + c$ and $g(\mathbf{x}) = \mathbf{x}^T B\mathbf{x} + b^T \cdot \mathbf{x} + d$ be two quadratic functions having symmetric matrices A and B. If $g(\mathbf{x})$ takes both positive and negative values and $B \neq 0$, then the following two statements are equivalent:*

1. $(\forall \mathbf{x} \in \mathbb{R}^n)\ g(\mathbf{x}) = 0 \implies f(\mathbf{x}) \geqslant 0$,
2. *There exists a $\mu \in \mathbb{R}$ such that $f(\mathbf{x}) + \mu g(\mathbf{x}) \geqslant 0,\ \forall \mathbf{x} \in \mathbb{R}^n$.*

Using this theorem and based on the approach described by Xia et al. [193], (NLP_i) can be reformulated as a Semidefinite Program and thus solved efficiently.

3.3.2 Necessary condition for KT-invexity

Theorem 3.7. *(Necessary condition) If (NLP) is KT-invex, then it is weakly boundary-invex.*

Proof. We will proceed by contradiction, assume that (NLP) is KT-invex but not weakly boundary-invex. The latter implies that for some i there exists a point $\mathbf{x}^* \in F$ which is a global minimiser of (NLP_i) and therefore its KKT point:

$$\begin{aligned}
&\nabla f(\mathbf{x}^*) = -\lambda_i \nabla g_i(\mathbf{x}^*),\\
&g_i(\mathbf{x}^*) = 0.
\end{aligned}$$

Moreover, $\mathbf{x}^*$ violating weak boundary-invexity implies that $\lambda_i < 0$. Let $\mu_i = -\lambda_i$. We can set $\mu_j = 0$ for $j = 1, \ldots, i-1, i+1, \ldots, m$ and obtain the following system:

$$\begin{aligned}
&\nabla f(\mathbf{x}^*) = \sum_{j=1}^{m} \mu_j \nabla g_j(\mathbf{x}^*),\\
&g_j(\mathbf{x}^*) \leqslant 0,\ \forall j = 1, \ldots, m,\\
&\mu_j \geqslant 0,\ \forall j = 1, \ldots, m,
\end{aligned}$$

implying that $\mathbf{x}^*$ is a KKT point of (NLP). Since $\mathbf{x}^*$ violates weak boundary-invexity, no other constraints are active at $\mathbf{x}^*$. Therefore there exists a point $\hat{\mathbf{x}}$ in the neighbourhood of $\mathbf{x}^*$, such that

$$g_i(\hat{\mathbf{x}}) = 0 \text{ and } \hat{\mathbf{x}} \in F.$$

The assumption that $\mathbf{x}^*$ violates weak boundary-invexity implies that $\mathbf{x}^*$ is a strict global minimiser in (NLP_i). Thus we have that $f(\mathbf{x}^*) < f(\hat{\mathbf{x}})$, which contradicts (NLP) being KT-invex.

□

3.3.3 Connection between boundary and interior optimality

Definition 3.5. *[171] A connected set is a set which cannot be represented as the union of two disjoint nonempty closed sets.*

Lemma 3.1. *Given a local maximiser $\mathbf{x}^* \in \mathbb{R}^n$ for (NLP), if F is connected then: If $\mathbf{x}^*$ is a global maximiser on ∂F then it is also a global maximiser for (NLP).*

Proof. Let $N(\mathbf{x}^*) \subseteq \mathbb{R}^n$ be any neighbourhood of $\mathbf{x}^*$ such that $f(\mathbf{x}) \leqslant f(\mathbf{x}^*)\ \forall \mathbf{x} \in N(\mathbf{x}^*) \cap F$. $N(\mathbf{x}^*)$ exists because $\mathbf{x}^*$ is a local maximum.

Consider a point $\hat{\mathbf{x}}$ such that $f(\hat{\mathbf{x}}) > f(\mathbf{x}^*)$. We will show that $\hat{\mathbf{x}}$ is infeasible. By concavity of f:

$$f(\hat{\mathbf{x}}) - f(\mathbf{x}^*) \leqslant (\nabla f(\mathbf{x}^*))^T \cdot (\hat{\mathbf{x}} - \mathbf{x}^*).$$

Since $f(\hat{\mathbf{x}}) > f(\mathbf{x}^*)$, we have that $(\nabla f(\mathbf{x}^*))^T \cdot (\hat{\mathbf{x}} - \mathbf{x}^*) > 0$ which implies that f locally increases at $\mathbf{x}^*$ in the direction $\hat{\mathbf{x}} - \mathbf{x}^*$. This and the continuity of f imply that there exists a point $\bar{\mathbf{x}} \in \overline{\mathbf{x}^*\hat{\mathbf{x}}} \cap N(\mathbf{x}^*)$ such that $f(\mathbf{x}^*) < f(\bar{\mathbf{x}}) < f(\hat{\mathbf{x}})$, where $\overline{\mathbf{x}^*\hat{\mathbf{x}}}$ denotes a segment between points $\mathbf{x}^*$ and $\hat{\mathbf{x}}$ in $\mathbb{R}^n$. Let $c = f(\bar{\mathbf{x}})$ and $L_c(f) = \{\mathbf{x} \mid f(\mathbf{x}) \geqslant c\}$. $L_c(f) \cap N(\mathbf{x}^*)$ is nonempty since it contains $\bar{\mathbf{x}}$. Note that $\hat{\mathbf{x}} \in L_c(f)$.

Since $f(\mathbf{x}) \leqslant f(\mathbf{x}^*)\ \forall \mathbf{x} \in \partial F$ and $f(\mathbf{x}) > f(\mathbf{x}^*)\ \forall \mathbf{x} \in \partial L_c(f)$, the two boundaries cannot have common points, i.e. $\partial L_c(f) \cap \partial F = \varnothing$. Given that $L_c(f)$ is convex [25] and F is connected, there are three possibilities:

1) If $F \subset L_c(f)$. Contradiction, since $\mathbf{x}^* \in F$ and $\mathbf{x}^* \notin L_c(f)$ given that $f(\mathbf{x}^*) < c$.

2) If $L_c(f) \subset F$. Given that $L_c(f) \cap N(\mathbf{x}^*)$ is nonempty, points in this intersection have a higher objective function value with respect to $\mathbf{x}^*$, belong to its neighbourhood and are feasible. This contradicts with $\mathbf{x}^*$ being a local maximiser.

3) If $F \cap L_c(f) = \varnothing$. In this case $\hat{\mathbf{x}} \notin F$.

We have proven that $\hat{\mathbf{x}} \notin F$ for any $\hat{\mathbf{x}}$ such that $f(\hat{\mathbf{x}}) > f(\mathbf{x}^*)$. Thus $\mathbf{x}^*$ is a global maximiser in F. □

3.3.4 Problems with two degrees of freedom

To the best of our knowledge, there are no polynomial-time verifiable necessary and sufficient conditions for checking KT-invexity even in two dimensions. In this work, we try to take a first step in this direction, showing that boundary-invexity is both necessary and sufficient while being efficiently verifiable. Even after restricting the problem to two degrees of freedom, the proof of sufficiency is not straightforward and requires an elaborate geometric reasoning. In the following sections, we try to break up our approach into various pieces, in the hope of making it easier for the reader.

We consider the following optimisation problem:

$$\begin{aligned} \max\ & f^0(\mathbf{x}) \\ \text{s.t. } & g_i^0(\mathbf{x}) \leqslant 0\ \forall i = 1, \ldots, m, \\ & h_j^0(\mathbf{x}) = 0\ \forall j = 1, \ldots, n-2, \end{aligned}$$

$$\mathbf{x} \in \mathbb{R}^n$$

and assume that $n-2$ variables can be projected out given the system of nonredundant $n-2$ linear equations $h_j^0(\mathbf{x}) = 0$ $(j = 1, \ldots, n-2)$. After projecting these variables out, the above can be expressed as a two-dimensional problem:

$$\begin{aligned} \max \;& f(x_1, x_2) \\ \text{s.t.} \;& g_i(x_1, x_2) \leqslant 0 \;\forall i = 1, \ldots, m, \\ & (x_1, x_2) \in \mathbb{R}^2. \end{aligned} \tag{NLP$_2$}$$

Definition 3.6. *[115] A real function f is said to be real analytic at $\mathbf{x}^0$ if it may be represented by a convergent power series in some neighbourhood of $\mathbf{x}^0$:*

$$f(\mathbf{x}) = \sum_{j=0}^{\infty} a_j (\mathbf{x} - \mathbf{x}^0)^j.$$

The function is said to be real analytic on a set $S \subset \mathbb{R}^n$ if it is real analytic at each $\mathbf{x}^0 \in S$.

Assumptions on (NLP$_2$). Throughout the chapter, we will assume that:

1. f is a concave real analytic function;
2. functions g_i $(i = 1, \ldots, m)$ are twice continuously differentiable;
3. F is connected and bounded;
4. the Linear Independence Constraint Qualification (LICQ) holds for all $\mathbf{x} \in \partial F$.

First let us prove a general result for problems with feasible sets in $\mathbb{R}^2$. For this proof we need to recall the definition of a redundant constraint:

Definition 3.7. *[182] A constraint is redundant if it can be removed from the problem without changing the feasible set.*

The next definition describes a property that distinguishes the constraints defining the boundary of F in the neighbourhood of a given feasible point.

Definition 3.8. *A constraint $g_i(\mathbf{x}) \leqslant 0$ is nonredundant around $\bar{\mathbf{x}} \in F$ if it satisfies the following conditions:*

1. *$g_i(\bar{\mathbf{x}}) = 0$,*
2. *there exists an $\epsilon > 0$ such that $g_i(\mathbf{x}) \leqslant 0$ is nonredundant in the neighbourhood of $\bar{\mathbf{x}}$ defined as $\{\mathbf{x}$ such that $||\mathbf{x} - \bar{\mathbf{x}}|| \leqslant \epsilon_0\}$ for all $\epsilon_0 \in (0, \epsilon]$.*

Lemma 3.2. *At most two constraints of (NLP$_2$) can be nonredundant around any $\bar{\mathbf{x}} \in \partial F$.*

Proof. Suppose that three constraints with indices $i, j, k \in 1, \ldots, m$ are nonredundant around $\bar{\mathbf{x}} \in \partial F$. Let $\mathbf{d}_i = (-\frac{\partial g_i}{\partial x_2}(\bar{\mathbf{x}}), \frac{\partial g_i}{\partial x_1}(\bar{\mathbf{x}}))^T$. One can see that $\mathbf{d}_i$ is a tangent direction from $\bar{\mathbf{x}}$ along $g_i(\mathbf{x}) = 0$ since it is orthogonal to $\nabla g_i(\bar{\mathbf{x}})$.

Since g_i is nonredundant around $\bar{\mathbf{x}}$, at least one of the two tangent directions $\mathbf{d}_i$ and $-\mathbf{d}_i$ is feasible. Assume without loss of generality that the feasible direction is $\mathbf{d}_i$. This implies that, in particular, $\mathbf{d}_i$ is feasible with respect to $g_j(\mathbf{x}) \leqslant 0$, that is, $\mathbf{d}_i^T \cdot \nabla g_j(\bar{\mathbf{x}}) \leqslant 0$. Since $\nabla g_i(\bar{\mathbf{x}})$ and $\nabla g_j(\bar{\mathbf{x}})$ are linearly independent, the inequality must be strict: $\mathbf{d}_i^T \cdot \nabla g_j(\bar{\mathbf{x}}) < 0$. This implies that direction $-\mathbf{d}_i$ is infeasible with respect to $g_j(\mathbf{x}) \leqslant 0$.

Letting $\mathbf{d}_j = (\frac{\partial g_j}{\partial x_2}(\bar{\mathbf{x}}), -\frac{\partial g_j}{\partial x_1}(\bar{\mathbf{x}}))^T$ and writing the inequality $\mathbf{d}_i^T \cdot \nabla g_j(\bar{\mathbf{x}}) < 0$ through components of the gradients: $-\frac{\partial g_i}{\partial x_2}(\bar{\mathbf{x}})\frac{\partial g_j}{\partial x_1}(\bar{\mathbf{x}}) + \frac{\partial g_i}{\partial x_1}(\bar{\mathbf{x}})\frac{\partial g_j}{\partial x_2}(\bar{\mathbf{x}}) < 0$, we obtain $\mathbf{d}_j \cdot \nabla g_i(\bar{\mathbf{x}}) < 0$.

In order for constraints $g_i(\mathbf{x}) \leqslant 0$, $g_j(\mathbf{x}) \leqslant 0$ to be feasible, both directions $\mathbf{d}_i$, $\mathbf{d}_j$ must be feasible with respect to $g_k(\mathbf{x}) \leqslant 0$ at $\bar{\mathbf{x}}$:

$$\mathbf{d}_i \cdot \nabla g_k(\bar{\mathbf{x}}) < 0,$$

$$\mathbf{d}_j \cdot \nabla g_k(\bar{\mathbf{x}}) < 0,$$

or, equivalently,

$$-\frac{\partial g_i}{\partial x_2}(\bar{\mathbf{x}})\frac{\partial g_k}{\partial x_1}(\bar{\mathbf{x}}) + \frac{\partial g_i}{\partial x_1}(\bar{\mathbf{x}})\frac{\partial g_k}{\partial x_2}(\bar{\mathbf{x}}) < 0, \tag{3.3}$$

$$\frac{\partial g_j}{\partial x_2}(\bar{\mathbf{x}})\frac{\partial g_k}{\partial x_1}(\bar{\mathbf{x}}) - \frac{\partial g_j}{\partial x_1}(\bar{\mathbf{x}})\frac{\partial g_k}{\partial x_2}(\bar{\mathbf{x}}) < 0. \tag{3.4}$$

Let $\mathbf{d}_k = (\frac{\partial g_k}{\partial x_2}(\bar{\mathbf{x}}), -\frac{\partial g_k}{\partial x_1}(\bar{\mathbf{x}}))^T$. Inequalities (3.3)-(3.4) imply that $-\mathbf{d}_k \cdot \nabla g_i(\bar{\mathbf{x}}) > 0$ and $\mathbf{d}_k \cdot \nabla g_j(\bar{\mathbf{x}}) > 0$. Thus both directions $\mathbf{d}_2$ and $-\mathbf{d}_2$ are infeasible. This contradicts the initial assumption that constraint $g_k(\mathbf{x}) \leqslant 0$ is nonredundant around $\bar{\mathbf{x}}$. □

The boundary-invexity models (NLP_i) for problem (NLP_2) are formulated as:

$$\begin{aligned} &\min \ f(x_1, x_2) \\ &s.t. \ g_i(x_1, x_2) = 0. \end{aligned} \tag{NLP_{2i}}$$

We will define a stronger version of the boundary-invexity property, which is both necessary and sufficient for KT-invexity of (NLP_2):

Definition 3.9. *(Boundary-invexity) Problem* (NLP_2) *is boundary-invex if at least one of the following holds for every KKT point* $\mathbf{x}^*$ *of* (NLP_{2i}) *for every* i *that corresponds to a nonconvex constraint:*

1. $\mathbf{x}^*$ *is infeasible for* (NLP_2),
2. *the Lagrange multiplier for* $\mathbf{x}^*$ *in* (NLP_{2i}) *is nonnegative,*
3. $\mathbf{x}^*$ *is a local maximum with respect to* (NLP_2).

Theorem 3.8. *(Necessary condition) If* (NLP$_2$) *is KT-invex, then it is boundary-invex.*

Proof. We will proceed by contradiction. Assume that (NLP$_2$) is KT-invex but not boundary-invex. The latter implies that there exists a point $\mathbf{x}^* \in F$ which violates boundary-invexity and it can be shown that it is a KKT point of (NLP$_2$) (the proof of this fact can be found in Theorem 3.7).

Since $\mathbf{x}^*$ violates boundary-invexity, it is not a local maximum with respect to (NLP$_2$). Therefore, it is not a global maximum, which contradicts with (NLP$_2$) being KT-invex.

□

3.3.5 Local optimality of KKT points

We start by rewriting Theorem 2.1 for the case of an inequality-constrained maximisation problem.

Theorem 3.9. *[150](Second-order sufficient conditions) Let $\mathbf{x}^*$ be a KKT point for problem* (NLP$_2$) *with Lagrange multiplier vectors μ, ν. Suppose that*

$$\mathbf{w}^T \nabla^2_{\mathbf{x}} L(\mathbf{x}^*, \mu)\mathbf{w} > 0 \ \forall \mathbf{w} \in C(\mathbf{x}^*, \mu), \ \ \mathbf{w} \neq 0,$$

where $L(\mathbf{x}, \mu) = -f(\mathbf{x}) + \sum\limits_{i=1}^{m} \mu_i g_i(\mathbf{x})$ is the Lagrangian function.

Then $\mathbf{x}^*$ is a strict local maximum in (NLP$_2$).

The following lemma proves local optimality of KKT points of boundary-invex problems.

Lemma 3.3. *Suppose that* (NLP$_2$) *is boundary-invex. Then every KKT point is a local maximum.*

Proof. Consider a KKT point $\mathbf{x}^*$. By Lemma 3.2, at most two constraints can be nonredundant around $\mathbf{x}^*$. Let these constraints be denoted as g_1 and g_2 and let the corresponding Lagrange multipliers be μ_1, μ_2.

1. If both $\mu_i > 0$, then the critical cone can be written as:

$$\mathbf{w} \in C(\mathbf{x}^*, \mu) \ \Leftrightarrow \ \begin{cases} (\nabla g_1(\mathbf{x}^*))^T \cdot \mathbf{w} = 0 \\ (\nabla g_2(\mathbf{x}^*))^T \cdot \mathbf{w} = 0. \end{cases}$$

   $\mathbf{w}$ is orthogonal to both $\nabla g_1(\mathbf{x}^*)$ and $\nabla g_2(\mathbf{x}^*)$. This is only possible if:

   (a) $\mathbf{w} = 0$ or

   (b) $\nabla g_1(\mathbf{x}^*)$ and $\nabla g_2(\mathbf{x}^*)$ are linearly dependent.

   In case (a), no nonzero vectors exist in the set $C(\mathbf{x}^*, \mu)$ and thus no vector $\mathbf{w}$ violates the conditions of Theorem 2.1. Then $\mathbf{x}^*$ is a strict local maximum. In case (b) LICQ is violated.

2. Suppose that $\mu_2 = 0$ and $\mu_1 > 0$. Then, by (2.1), $\nabla f(\mathbf{x}^*) = \mu_1 \nabla g_1(\mathbf{x}^*)$. Then the following cases are possible:

   (a) g_1 is convex. Since (2.2)-(2.5) are satisfied, $\mathbf{x}^*$ is a KKT point for a problem of maximizing a concave function f over a convex region $g_1(\mathbf{x}) \leqslant 0$. Then it is a local maximum for this problem and, since it is a relaxation of (NLP_2), a local maximum for (NLP_2).

   (b) g_1 is nonconvex. Setting $\lambda_1 = -\mu_1$, we get $\nabla f(\mathbf{x}^*) = -\lambda_1 \nabla g_1(\mathbf{x}^*)$, $\lambda_1 < 0$. (2.5) implies that $g_1(\mathbf{x}^*) = 0$. Then $\mathbf{x}^*$ is a KKT point for (NLP_{2i}) with a negative Lagrange multiplier which is feasible for (NLP_2). Since (NLP_2) is boundary-invex, $\mathbf{x}^*$ is a local maximum.

3. $\mu_1 = \mu_2 = 0$. Then $\mathbf{x}^*$ is the unconstrained global maximum of f and thus a maximum for (NLP_2).

□

3.4 Pseudo-Scalar Product

Definition 3.10. *[151] Given two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$ define their pseudo-scalar product to be*

$$\mathbf{x} \times \mathbf{y} = x_1 y_2 - x_2 y_1.$$

The sign of $\mathbf{x} \times \mathbf{y}$ has a geometric interpretation. If $\mathbf{x} \times \mathbf{y} > 0$, then the shortest angle at which $\mathbf{x}$ has to be rotated for it to become co-directional with $\mathbf{y}$ corresponds to a counter-clockwise rotation. If $\mathbf{x} \times \mathbf{y} < 0$, then such an angle corresponds to a clockwise rotation. If $\mathbf{x} \times \mathbf{y} = 0$, the vectors are parallel.

Definition 3.11 (Tangent vector)**.** *[1] Given a parametrisation $(x_1(t), x_2(t))$ of a curve $g(x_1, x_2) = 0$, the vector $(x_1'(t), x_2'(t))^T$ is said to be its tangent vector.*

Tangent vectors are orthogonal to gradient vectors. This can be proven using the chain differentiation rule:

$$g(x_1(t), x_2(t)) = 0 \;\Rightarrow\; \frac{\partial g}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial g}{\partial x_2}\frac{\partial x_2}{\partial t} = (x_1'(t), x_2'(t)) \cdot \nabla g(x_1, x_2) = 0.$$

Lemma 3.4. *Given a differentiable function $g : \mathbb{R}^2 \to \mathbb{R}$, a point $\mathbf{y} = (y_1, y_2)$ such that $g(\mathbf{y}) = 0$, the vector $\left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right)$ is the tangent vector to the curve $g(\mathbf{x}) = 0$ at point $\mathbf{y}$.*

Proof. Considering the dot product,

$$\left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right) \cdot \nabla g(\mathbf{y}) = \left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right) \cdot \left(\frac{\partial g}{\partial x_1}(\mathbf{y}), \frac{\partial g}{\partial x_2}(\mathbf{y})\right)^T =$$

$$= -\frac{\partial g}{\partial x_2}(\mathbf{y})\frac{\partial g}{\partial x_1}(\mathbf{y}) + \frac{\partial g}{\partial x_1}(\mathbf{y})\frac{\partial g}{\partial x_2}(\mathbf{y}) = 0,$$

the vector $\left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right)$ is orthogonal to the gradient and thus a tangent to the curve $g(\mathbf{x}) = 0$ at the point $\mathbf{y}$.

□

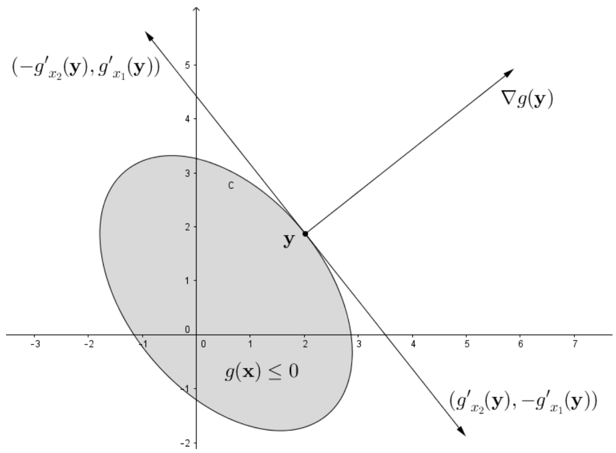


Figure 3.1: Tangent vectors

Definition 3.12. *The positive (resp. negative) direction of moving along the curve* $g(\mathbf{x}) = 0$ *is the direction corresponding to the vector* $\left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right)$ *(resp.* $\left(\frac{\partial g}{\partial x_2}(\mathbf{y}), -\frac{\partial g}{\partial x_1}(\mathbf{y})\right)$*).*

Lemma 3.5. *Consider differentiable functions* $f : \mathbb{R}^2 \to \mathbb{R}$ *and* $g : \mathbb{R}^2 \to \mathbb{R}$*. We have* $\nabla f(\mathbf{y}) \times \nabla g(\mathbf{y}) \geqslant 0$ *(resp.* $\nabla f(\mathbf{y}) \times \nabla g(\mathbf{y}) \leqslant 0$*) if and only if* f *is nonincreasing (resp. nondecreasing) when moving along the curve* $g(\mathbf{x}) = 0$ *in the positive direction.*

Proof. We will prove the case where f is nonincreasing.

Consider the directional derivative of f with respect to the tangent vector at point $\mathbf{y}$:

$$\frac{\partial f}{\partial(-g'_{x_2}(\mathbf{y}), g'_{x_1}(\mathbf{y}))}(\mathbf{y}) = \left(-\frac{\partial g}{\partial x_2}(\mathbf{y}), \frac{\partial g}{\partial x_1}(\mathbf{y})\right) \cdot \nabla f(\mathbf{y}) =$$

$$-\frac{\partial f}{\partial x_1}(\mathbf{y})\frac{\partial g}{\partial x_2}(\mathbf{y}) + \frac{\partial f}{\partial x_2}(\mathbf{y})\frac{\partial g}{\partial x_1}(\mathbf{y}) = -\nabla f(\mathbf{y}) \times \nabla g(\mathbf{y}) \leqslant 0,$$

and this implies that the pseudo-scalar product being nonnegative at $\mathbf{y}$ is equivalent to f being nonincreasing on $g(\mathbf{x}) = 0$ at $\mathbf{y}$.

□

3.4.1 Reformulation of the KKT conditions

Now we shall establish a connection between the KKT conditions and the sign of the pseudo-scalar products corresponding to the gradient vectors.

Lemma 3.6. *Consider a point* $\mathbf{x}^* \in F$ *with two active nonredundant constraints* $g_1(\mathbf{x}) \leqslant 0$ *and* $g_2(\mathbf{x}) \leqslant 0$ *such that* $\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) > 0$*.* $\mathbf{x}^*$ *is a KKT point if and only if*

$$\begin{aligned} \nabla f(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) \geqslant 0, \\ \nabla f(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) \leqslant 0. \end{aligned}$$

Proof. By KKT conditions (2.1)-(2.5), there exist μ_1, μ_2 such that the following holds:

$$\begin{cases} \mu_1 \frac{\partial g_1}{\partial x_1}(\mathbf{x}^*) + \mu_2 \frac{\partial g_2}{\partial x_1}(\mathbf{x}^*) = \frac{\partial f}{\partial x_1}(\mathbf{x}^*), \\ \mu_1 \frac{\partial g_1}{\partial x_2}(\mathbf{x}^*) + \mu_2 \frac{\partial g_2}{\partial x_2}(\mathbf{x}^*) = \frac{\partial f}{\partial x_2}(\mathbf{x}^*), \\ \mu_1, \mu_2 \geqslant 0. \end{cases}$$

From this system we can find μ_1, μ_2:

$$\mu_1 = \frac{\frac{\partial f}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_2}{\partial x_2}(\mathbf{x}^*) - \frac{\partial g_2}{\partial x_1}(\mathbf{x}^*)\frac{\partial f}{\partial x_2}(\mathbf{x}^*)}{\frac{\partial g_1}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_2}{\partial x_2}(\mathbf{x}^*) - \frac{\partial g_2}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_1}{\partial x_2}(\mathbf{x}^*)} = \frac{\nabla f(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*)}{\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*)},$$

$$\mu_2 = \frac{\frac{\partial g_1}{\partial x_1}(\mathbf{x}^*)\frac{\partial f}{\partial x_2}(\mathbf{x}^*) - \frac{\partial f}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_1}{\partial x_2}(\mathbf{x}^*)}{\frac{\partial g_1}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_2}{\partial x_2}(\mathbf{x}^*) - \frac{\partial g_2}{\partial x_1}(\mathbf{x}^*)\frac{\partial g_1}{\partial x_2}(\mathbf{x}^*)} = \frac{\nabla g_1(\mathbf{x}^*) \times \nabla f(\mathbf{x}^*)}{\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*)}.$$

$\mu_1, \mu_2 \geqslant 0$ is equivalent to

$$\begin{aligned} \nabla f(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) \geqslant 0, \\ \nabla f(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) \leqslant 0. \end{aligned}$$

□

3.5 Parametrisation of the Boundary of F

Given a nonempty connected set F and a real variable $t \in [0, T]$, where $T \in \mathbb{R},\ T > 0$, define a parametrisation $\gamma : \mathbb{R} \to \mathbb{R}^2$ of ∂F such that $\gamma(0) = \gamma(T)$ and the direction of increase of t corresponds to the positive direction of moving along the boundary. Then

$$\begin{aligned} \gamma'_-(t) &= \left(-\frac{\partial g_{i^-(t)}}{\partial x_2}(\gamma(t)), \frac{\partial g_{i^-(t)}}{\partial x_1}(\gamma(t))\right)^T, \\ \gamma'_+(t) &= \left(-\frac{\partial g_{i^+(t)}}{\partial x_2}(\gamma(t)), \frac{\partial g_{i^+(t)}}{\partial x_1}(\gamma(t))\right)^T, \end{aligned}$$

where $i^-(t)$ and $i^+(t)$ are indices of constraints that are nonredundant around $\gamma(t)$. Let $i^-(t) = i(t - \epsilon)$ and $i^+(t) = i(t + \epsilon)\ \forall \epsilon \in (0, \epsilon_0)$ for some $\epsilon_0 > 0$. If only one constraint is nonredundant around $\gamma(t)$, then $i^-(t) = i^+(t) = i(t)$ and $\gamma'_-(t) = \gamma'_+(t) = \gamma'(t)$.

We will also require that for any $t_1 \in (0,T),\ t_2 \in (0,T) \mid t_1 \neq t_2$ these vectors are not equal: $(\gamma(t_1)^T, \gamma'_-(t_1)^T, \gamma'_+(t_1)^T) \neq (\gamma(t_2)^T, \gamma'_-(t_2)^T, \gamma'_+(t_2)^T)$. This is needed to ensure that the parametrisation represents exactly one "walk" around the curve.

Let $\gamma^r(t)$ be the reversed direction parametrisation of ∂F:

$$\gamma^{r\prime}_-(t) = \left(\frac{\partial g_{i^{r-}(t)}}{\partial x_2}(\gamma(t)), -\frac{\partial g_{i^{r-}(t)}}{\partial x_1}(\gamma(t))\right)^T,$$
$$\gamma^{r\prime}_+(t) = \left(\frac{\partial g_{i^{r+}(t)}}{\partial x_2}(\gamma(t)), -\frac{\partial g_{i^{r+}(t)}}{\partial x_1}(\gamma(t))\right)^T,$$

where $i^{r-}(t)$, $i^{r+}(t)$ are defined in a similar way to the indices in the direct parametrisation.

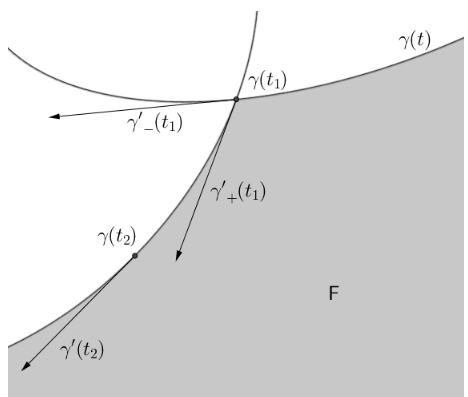


Figure 3.2: Parametrisation of the boundary of the feasible region

In the following Lemma, we show that γ does not intersect itself.

Lemma 3.7. *Consider two distinct values t_1 and t_2 of parameter t such that $0 < t_1 < t_2 < T$, then $\gamma(t_1) \neq \gamma(t_2)$.*

Proof. We will proceed by contradiction, suppose that there exist numbers t_1, t_2 such that $\gamma(t_1) = \gamma(t_2) = \mathbf{y}$ and $0 < t_1 < t_2 < T$. Let $j = i^+(t_1)$ and $k = i^-(t_2)$. By Lemma 3.2, either one or two constraints can be nonredundant around $\mathbf{y}$.

If one constraint is nonredundant around $\mathbf{y}$, then $j = k$ and $\gamma'_-(t_1) = \gamma'_-(t_2)$ and $\gamma'_+(t_1) = \gamma'_+(t_2)$. This contradicts the requirement that $(\gamma(t_1)^T, \gamma'_-(t_1)^T, \gamma'_+(t_1)^T) \neq (\gamma(t_2)^T, \gamma'_-(t_2)^T, \gamma'_+(t_2)^T)\ \ \forall t_1, t_2 \in (0,T)$.

Now suppose that $j \neq k$ and two constraints are nonredundant around $\mathbf{y}$. Since $(\gamma(t_1)^T, \gamma'_-(t_1)^T, \gamma'_+(t_1)^T) \neq (\gamma(t_2)^T, \gamma'_-(t_2)^T, \gamma'_+(t_2)^T)$, we have that $i^-(t_1) \neq k$ and $j \neq i^+(t_2)$. Therefore $j = i^-(t_1) = i(t_1)$ and $k = i^+(t_2) = i(t_2)$. Consider the product $(\nabla g_j(\mathbf{y}))^T \cdot \gamma'(t_2)$. Two cases are possible:

1. $(\nabla g_j(\mathbf{y}))^T \cdot \gamma'(t_2) = 0$. Then

$$(g_j)'_{x_1}(\mathbf{y})(g_k)'_{x_2}(\mathbf{y}) - (g_j)'_{x_2}(\mathbf{y})(g_k)'_{x_1}(\mathbf{y})$$
$$= (\nabla g_j(\mathbf{y}))^T \cdot ((g_k)'_{x_2}(\mathbf{y}), -(g_k)'_{x_1}(\mathbf{y}))^T = -(\nabla g_j(\mathbf{y}))^T \cdot \gamma'(t_2) = 0. \tag{3.5}$$

If $(g_k)'_{x_1}(\mathbf{y}) = 0$, then $(g_j)'_{x_1}(\mathbf{y})(g_k)'_{x_2}(\mathbf{y}) = 0$. By LICQ, $\nabla g_k(\mathbf{y}) \neq 0$, which implies that $(g_k)'_{x_2}(\mathbf{y}) \neq 0$. Therefore, in order for (3.5) to hold, $(g_j)'_{x_1}(\mathbf{y})$ must be zero. We have two vectors that have zero x_1-components and nonzero x_2-components and, thus, are linearly dependent. This contradicts the assumption that LICQ holds for all boundary points of F.

If $(g_k)'_{x_1}(\mathbf{y}) \neq 0$, we can divide by this number:

$$(g_j)'_{x_2}(\mathbf{y}) = \frac{(g_j)'_{x_1}(\mathbf{y})(g_k)'_{x_2}(\mathbf{y})}{(g_k)'_{x_1}(\mathbf{y})}.$$

If $c = -\frac{(g_j)'_{x_1}(\mathbf{y})}{(g_k)'_{x_1}(\mathbf{y})}$, we have that

$$c\nabla g_k(\mathbf{y}) = -\frac{(g_j)'_{x_1}(\mathbf{y})}{(g_k)'_{x_1}(\mathbf{y})} \begin{pmatrix} (g_k)'_{x_1}(\mathbf{y}) \\ (g_k)'_{x_2}(\mathbf{y}) \end{pmatrix} = -\begin{pmatrix} (g_j)'_{x_1}(\mathbf{y}) \\ \frac{(g_k)'_{x_2}(\mathbf{y})(g_j)'_{x_1}(\mathbf{y})}{(g_k)'_{x_1}(\mathbf{y})} \end{pmatrix}$$
$$= -\begin{pmatrix} (g_j)'_{x_1}(\mathbf{y}) \\ (g_j)'_{x_2}(\mathbf{y}) \end{pmatrix}$$

and

$$\nabla g_j(\mathbf{y}) + c\nabla g_k(\mathbf{y}) = \nabla g_j(\mathbf{y}) - \nabla g_j(\mathbf{y}) = 0.$$

This violates LICQ.

2. $(\nabla g_j(\mathbf{y}))^T \cdot \gamma'(t_2) \neq 0$. This product can be interpreted as the directional derivative of g_j with respect to $\gamma'(t_2)$. Note that $g_j(\mathbf{y}) = 0$. Since the directional derivative is nonzero and $\gamma'(t_2)$ locally approximates $\gamma(t)$, then g_j changes sign on $\gamma(t)$ at t_2. Then we either have $g_j(\gamma(t_2 - \epsilon)) < 0$ and $g_j(\gamma(t_2 + \epsilon)) > 0$, or $g_j(\gamma(t_2 - \epsilon)) > 0$. In both cases there exist infeasible points on $\gamma(t)$. But since F is a closed set, $\partial F \in F$ and all points $\mathbf{x} = \gamma(t),\ t \in [0, T]$ are feasible. Contradiction.

□

Lemma 3.8. *Consider a boundary point $\mathbf{y} = \gamma(t^y)$. If there exist two constraints that are nonredundant around $\mathbf{y}$, then $\nabla g_{i^-(t^y)}(\mathbf{y}) \times \nabla g_{i^+(t^y)}(\mathbf{y}) > 0$.*

Proof. Consider the vector $\gamma'_+(t^y)$, which is the tangent vector to $g_{i^+(t^y)}$ at point $\mathbf{y}$. By definition of i^+, constraint $g_{i^+(t^y)}$ is active and nonredundant on $\gamma(t)$ in some right neighbourhood of t^y. Then the tangent is a feasible direction at $\mathbf{y}$ with respect to constraint $g_{i^-(t^y)}(\mathbf{x}) \leqslant 0$. This can be written as:

$$(\nabla g_{i^-(t^y)}(\mathbf{y}))^T \cdot \gamma'_+(t^y) \leqslant 0.$$

Or, equivalently:

$$\left(\frac{\partial g_{i^-(t^y)}}{\partial x_1}(\mathbf{y}), \frac{\partial g_{i^-(t^y)}}{\partial x_2}(\mathbf{y})\right) \cdot \left(-\frac{\partial g_{i^+(t^y)}}{\partial x_2}(\mathbf{y}), \frac{\partial g_{i^+(t^y)}}{\partial x_1}(\mathbf{y})\right)^T \leqslant 0 \;\Leftrightarrow$$

$$-\left(\frac{\partial g_{i^-(t^y)}}{\partial x_1}(\mathbf{y})\frac{\partial g_{i^+(t^y)}}{\partial x_2}(\mathbf{y}) - \frac{\partial g_{i^-(t^y)}}{\partial x_2}(\mathbf{y})\frac{\partial g_{i^+(t^y)}}{\partial x_1}(\mathbf{y})\right) \leqslant 0 \;\Leftrightarrow$$

$$\left(\frac{\partial g_{i^-(t^y)}}{\partial x_1}(\mathbf{y})\frac{\partial g_{i^+(t^y)}}{\partial x_2}(\mathbf{y}) - \frac{\partial g_{i^-(t^y)}}{\partial x_2}(\mathbf{y})\frac{\partial g_{i^+(t^y)}}{\partial x_1}(\mathbf{y})\right) \geqslant 0 \;\Leftrightarrow$$

$$\nabla g_{i^-(t^y)}(\mathbf{y}) \times \nabla g_{i^+(t^y)}(\mathbf{y}) \geqslant 0.$$

If $\nabla g_{i^-(t^y)}(\mathbf{y}) \times \nabla g_{i^+(t^y)}(\mathbf{y}) = 0$, then LICQ is violated at point $\mathbf{y}$:

$$\nabla g_{i^-(t^y)}(\mathbf{y}) + c\nabla g_{i^+(t^y)}(\mathbf{y}) = 0 \text{ if } c = \left(\frac{\partial g_{i^-(t^y)}}{\partial x_1}(\mathbf{y})\right) / \left(\frac{\partial g_{i^+(t^y)}}{\partial x_1}(\mathbf{y})\right).$$

Thus only strict inequality is possible:

$$\nabla g_{i^-(t^y)}(\mathbf{y}) \times \nabla g_{i^+(t^y)}(\mathbf{y}) > 0.$$

□

3.6 Splitting the Space in Two

3.6.1 Behaviour of a concave function on a line

First we will cite a general result for real analytic functions:

Lemma 3.9. *[115] If f and g are real analytic functions of one variable on an open interval U and there is an open set $W \subset U$ such that*

$$f(x) = g(x) \;\forall x \in W,$$

then

$$f(x) = g(x) \;\forall x \in U.$$

Let $f : \mathbb{R}^2 \to \mathbb{R}$ be a real analytic concave function. Consider a linear function $l(x_1, x_2) = ax_1 + bx_2 + c$. Let $\mathbf{y}$ be a point such that $l(\mathbf{y}) = 0$. We will define two rays:

Definition 3.13. *$r^d(\mathbf{y})$ is the ray lying on the line $l(\mathbf{x}) = 0$ starting at $\mathbf{y}$ and pointing in the locally decreasing direction of f.*

Definition 3.14. *$r^i(\mathbf{y})$ is the ray lying on the line $l(\mathbf{x}) = 0$ starting at $\mathbf{y}$ and pointing in the locally increasing direction of f.*

Let $\mathbf{x}_l^{max}$ be a point maximizing f subject to $l(\mathbf{x}) = 0$.

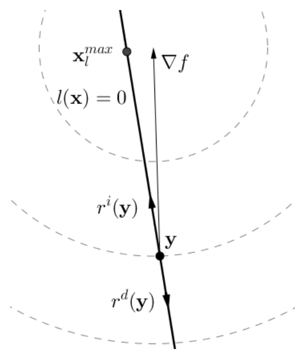


Figure 3.3: Rays $r^i(\mathbf{y})$ and $r^d(\mathbf{y})$

Lemma 3.10. *If a concave real analytic function f is not identically constant on $l(\mathbf{x}) = 0$ then it is strictly decreasing on $r^d(\mathbf{y})$.*

Proof. Consider two points $\mathbf{x}^1, \mathbf{x}^2 \in r^d(\mathbf{y})$ such that $||\mathbf{x}^2 - \mathbf{y}|| > ||\mathbf{x}^1 - \mathbf{y}||$. Since f is locally decreasing at $\mathbf{y}$ in the direction of $r^d(\mathbf{y})$, $(\nabla f(\mathbf{y}))^T \cdot (\mathbf{x}^1 - \mathbf{y}) \leqslant 0$. By concavity of f we have:

$$f(\mathbf{x}^1) - f(\mathbf{y}) \leqslant (\nabla f(\mathbf{y}))^T \cdot (\mathbf{x}^1 - \mathbf{y}) \;\Rightarrow\; f(\mathbf{x}^1) - f(\mathbf{y}) \leqslant 0 \;\Rightarrow\; f(\mathbf{y}) - f(\mathbf{x}^1) \geqslant 0.$$

Using the concavity of f again, we get:

$$f(\mathbf{y}) - f(\mathbf{x}^1) \leqslant (\nabla f(\mathbf{x}^1))^T \cdot (\mathbf{y} - \mathbf{x}^1) \;\Rightarrow\; (\nabla f(\mathbf{x}^1))^T \cdot (\mathbf{y} - \mathbf{x}^1) \geqslant 0$$
$$\Rightarrow\; (\nabla f(\mathbf{x}^1))^T \cdot (\mathbf{x}^2 - \mathbf{x}^1) \leqslant 0.$$

Repeating the same reasoning for $\mathbf{x}^1$ and $\mathbf{x}^2$ as for $\mathbf{y}$ and $\mathbf{x}^1$, we can show that $f(\mathbf{x}^2) \leqslant f(\mathbf{x}^1)$.

Since $f(x_1, x_2)$ is real analytic, so is $f(x_1, -\frac{ax_1+c}{b})$, which is the function of one variable x_1 and represents the behaviour of f on $l(\mathbf{x}) = 0$. Since $f(x_1, -\frac{ax_1+c}{b})$ is not identically constant, by Lemma 3.9 no interval exists where it is constant. Then strict inequality holds: $f(\mathbf{x}^2) < f(\mathbf{x}^1)$.

□

3.6.2 Boundary optimality on a half-plane

Let $\hat{\mathbf{x}} = \gamma(\hat{t})$ be a point on the boundary of F. In this section we will assume that for the parametrisation $\gamma(t)$ defined in Section 3.5, $f(\gamma(t))$ is nonincreasing as a function of t on some interval $[\hat{t}, \hat{t} + \epsilon]$, where $\epsilon > 0$. Otherwise, similar results can be proven for the reverse direction parametrisation $\gamma^r(t)$.

Definition 3.15. *[137] A path in $\mathbb{R}^n$ is a continuous function mapping every point in the unit interval $[0, 1]$ to a point in $\mathbb{R}^n$:*

$$\rho : [0, 1] \to \mathbb{R}^n.$$

Consider a linear function $l : \mathbb{R}^2 \to \mathbb{R}$ such that $l(\hat{\mathbf{x}}) = 0$, $\nabla f(\hat{\mathbf{x}}) \times \nabla l < 0$ and $\nabla l \times \nabla g_{i^+(\hat{t})}(\hat{\mathbf{x}}) > 0$ (these inequalities would be reversed for the case where $f(\gamma^r(t))$ is nonincreasing and $g_{i^+(\hat{t})}$ would be replaced with $g_{i^{r+}(\hat{t})}$). By Lemma 3.3 this and the assumption on monotonicity of f on γ imply that $\hat{\mathbf{x}}$ is a KKT point of the (NLP$_2$) problem with an additional constraint $l(\mathbf{x}) \leqslant 0$. Let $t^1 > \hat{t}$ be a parameter value corresponding to the point where $\gamma(t)$ first crosses the line $l(\mathbf{x}) = 0$ after $\hat{t}$:

$$t^1 = \begin{cases} \min\{t > \hat{t} \mid l(\gamma(t)) = 0\} \text{ if such } t \text{ exists}, \\ \infty \text{ otherwise.} \end{cases}$$

t^1 exists if F is bounded.

Define the optimisation problem

$$\begin{aligned} \max\ & f(x_1, x_2) \\ \text{s.t.}\ & g_i(x_1, x_2) \leqslant 0\ \forall i = 1, \ldots, m, \\ & l(x_1, x_2) \leqslant 0, \\ & (x_1, x_2) \in \mathbb{R}^2. \end{aligned} \tag{NLP$_l$}$$

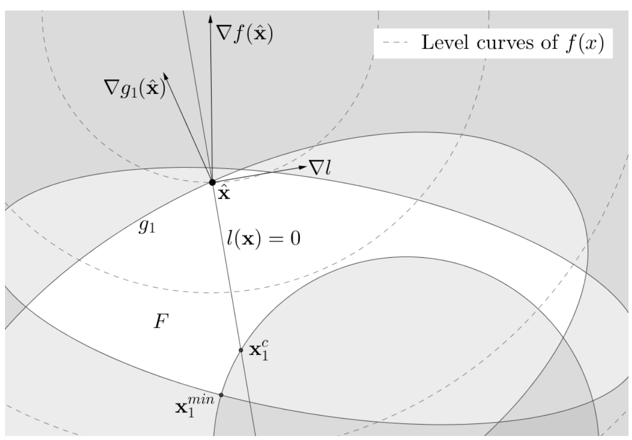


Figure 3.4: An example problem for Lemma 3.11

Lemma 3.11. *Given $\gamma(t)$, a parametrisation of ∂F in (NLP$_2$) and given a linear function l, if (NLP$_2$) is boundary-invex and $\hat{\mathbf{x}}$ is a KKT point of (NLP$_l$), then $f(\gamma(t)) \leqslant f(\gamma(\hat{t}))\ \forall t \in [\hat{t}, t^1]$.*

Proof. Let t^1_{min} denote the parameter value corresponding to the point where $f(\gamma(t))$ starts increasing as a function of t:

$$\begin{cases} t^1_{min} > \hat{t}, \\ (\nabla f(\gamma(t)) \times \nabla g_{i(t)}(\gamma(t))) \geqslant 0 \ \forall t \in (\hat{t}, t^1_{min}), \\ (\nabla f(\gamma(t)) \times \nabla g_{i(t)}(\gamma(t))) < 0 \ \forall t \in (t^1_{min}, t^1_{min} + \epsilon) \text{ for some } \epsilon > 0. \end{cases}$$

t^1_{min} does not exist only if $f(\gamma(t))$ is non-increasing for all $t \in [0, T]$. In this case, the statement of this lemma holds trivially. Otherwise, let $\mathbf{x}^1_{min} = \gamma(t^1_{min})$.

If $t^1_{min} = t^1$, then for all $\hat{t} \leqslant t \leqslant t^1$ the inequality $f(\gamma(t)) \leqslant f(\gamma(\hat{t}))$ is satisfied and the statement of the lemma holds. Now suppose that $t^1_{min} < t^1$.

Consider the set

$$L_1 = \left\{ \mathbf{x} \;\middle|\; \begin{cases} l(\mathbf{x}) \leqslant 0 \\ f(\mathbf{x}) \geqslant f(\mathbf{x}^1_{min}) \end{cases} \right\}$$

and the curve $\gamma_1(t) = \gamma(t),\ t \in [\hat{t}, t^1_{min})$. F is connected, $\gamma_1(t)$ is piecewise-continuous, and $\gamma_1(\hat{t}) = \hat{\mathbf{x}}$ lies on the line $l(\mathbf{x}) = 0$ and $\gamma_1(t^1_{min})$ lies on the curve $f(\mathbf{x}) = f(\mathbf{x}^1_{min})$, and these are the only points of intersection of the curve and the boundary of L_1. Thus $\gamma_1(t)$ is dividing L_1 into two connected sets. We will denote the set where all points in the neighbourhood of $\gamma_1(t)$ are feasible as S_1.

We know that, by definition of S_1, all points on its boundary belong to one of the following sets:

1. The level curve $f(\mathbf{x}) = f(\mathbf{x}^1_{min})$. By definition of $\mathbf{x}^1_{min}$ we have that $f(\mathbf{x}^1_{min}) \leqslant f(\hat{\mathbf{x}})$.

2. The curve $\gamma_1(t)$. By definition of $\gamma_1(t)$ and t^1_{min}, $f(\mathbf{x}) \leqslant f(\hat{\mathbf{x}})\ \forall \mathbf{x} \in \gamma_1(t)$.

3. The line $l(\mathbf{x}) = 0$. The assumptions on l imply that only the negative direction of moving along the boundary of $l(\mathbf{x}) \leqslant 0$ is feasible with respect to $g_{i^+(\hat{t})} \leqslant 0$ and f locally decreases in this direction. Together with the fact that $\hat{\mathbf{x}}$ is the only point where $\gamma_1(t)$ crosses the line, we have that points $\mathbf{x}$ in S_1 for which $l(\mathbf{x}) = 0$ lie on the ray $r^d(\hat{\mathbf{x}})$ and, by Lemma 3.10, satisfy $f(\mathbf{x}) \leqslant f(\hat{\mathbf{x}})$.

Thus $f(\mathbf{x}) \leqslant f(\hat{\mathbf{x}})\ \forall \mathbf{x} \in \partial S_1$. By Lemma 3.3, $\hat{\mathbf{x}}$ is a local maximum in S_1 and thus, by Lemma 3.1, $f(\mathbf{x}) \leqslant f(\hat{\mathbf{x}})\ \forall \mathbf{x} \in S_1$.

The points following $\gamma(t^1_{min})$ are in S_1

We will say that a path ρ starting at some point $\mathbf{x}^s \in \gamma_1(t)$ is S_1-feasible if $\mathbf{x} \in \rho \implies \mathbf{x} \in S_1$ and for all constraints g_i that are active on $\gamma_1(t)$, $g_i(\mathbf{x}) < 0$ for all $\mathbf{x}$ on ρ in some neighbourhood of $\mathbf{x}^s$ excluding $\mathbf{x}^s$ itself.

Consider a neighbourhood $N(\mathbf{x}^1_{min})$ such that only constraints $g_{i^-(t^1_{min})}$ and $g_{i^+(t^1_{min})}$ are nonredundant in it.

Let $t^- = t^1_{min} - \epsilon$ and $t^+ = t^1_{min} + \epsilon$ for some $\epsilon > 0$ and let:

$$\mathbf{x}^- = \gamma(t^-),\ \mathbf{x}^+ = \gamma(t^+),$$

$$\phi^- = g_{i^-(t^1_{min})},\ \phi^+ = g_{i^+(t^1_{min})}.$$

We will show that there exists an ϵ_0 such that for all $\epsilon < \epsilon_0$ the segment connecting $\gamma(t^-)$ and $\gamma(t^+)$ satisfies the conditions defined for the path ρ.

By Lemma 3.2, two cases are possible:

1. One constraint is nonredundant around $\mathbf{x}^1_{min}$.

   Define $\phi = \phi^- = \phi^+$.

   In this case $\mathbf{x}^1_{min}$ is a local minimum of f on $\phi(\mathbf{x}) = 0$. Then ϕ is either linear, concave or convex in some neighbourhood $N(\mathbf{x}^1_{min})$.

   If ϕ is linear, f being nonincreasing at $\gamma(t^-)$ implies that it stays nonincreasing on the segment of γ where ϕ is active. This contradicts with f changing monotonicity on such a segment.

   If ϕ is concave in $N(\mathbf{x}^1_{min})$, then $\mathbf{x}^1_{min}$ violates boundary-invexity of (NLP$_2$). Indeed, this point is a KKT point for (NLP$_{2i}$) with a negative Lagrange multiplier and not a local maximum for (NLP$_2$).

   Then ϕ can only be convex in $N(\mathbf{x}^1_{min})$.

   Since $\mathbf{x}^+$ is feasible and belongs to the neighbourhood of $\mathbf{x}^1_{min}$, then $\phi(\mathbf{x}) \leqslant 0\ \forall \mathbf{x} \in \overline{\mathbf{x}^-\mathbf{x}^+}$ and $\phi(\mathbf{x}) < 0$ for all $\mathbf{x}$ on this segment excluding $\mathbf{x}^-$ and $\mathbf{x}^+$. Hence $\overline{\mathbf{x}^-\mathbf{x}^+}$ is an S_1-feasible path and $\mathbf{x}^+ \in S_1$.

2. Two constraints are nonredundant around $\mathbf{x}^1_{min}$.

   By Lemma 3.8, $\nabla\phi^-(\mathbf{x}^1_{min}) \times \nabla\phi^+(\mathbf{x}^1_{min}) > 0$. By definition of the pseudo-scalar product, this is equivalent to:

$$(\nabla\phi^-(\mathbf{x}^1_{min}))^T \cdot \gamma'_+(t^1_{min}) < 0.$$

   This product can be interpreted as the directional derivative of ϕ^- with respect to the vector $\gamma'_+(t^1_{min})$. Observe that $\gamma'_+(t^1_{min})$ shows how $\mathbf{x}$ behaves on $\gamma(t)$ when small changes to t are made. Therefore, the above inequality implies that there exists ϵ_0 such that for any $\epsilon < \epsilon_0$ the following holds:

$$(\nabla\phi^-(\mathbf{x}^1_{min}))^T \cdot (\mathbf{x}^+ - \mathbf{x}^1_{min}) < 0.$$

   Since all constraints are twice continuously differentiable, $\nabla\phi^-(\mathbf{x}) \cdot (\mathbf{x}^+ - \mathbf{x})$ is a differentiable function of $\mathbf{x}$. Thus there exists a neighbourhood $N(\mathbf{x}^1_{min})$ where this function stays negative. We can choose ϵ_0 such that $\mathbf{x}^- \in N(\mathbf{x}^1_{min})\ \forall\epsilon < \epsilon_0$ and:

$$(\nabla\phi^-(\mathbf{x}^-))^T \cdot (\mathbf{x}^+ - \mathbf{x}^-) < 0.$$

Therefore $\phi^-(\mathbf{x}) < 0$ in some neighbourhood of $\mathbf{x}^-$ excluding $\mathbf{x}^-$ and there exists ϵ_0 such that $\phi^-(\mathbf{x}) \leqslant 0 \ \forall \mathbf{x} \in \overline{\mathbf{x}^-\mathbf{x}^+}$ if $\epsilon < \epsilon_0$. Thus the segment $\overline{\mathbf{x}^-\mathbf{x}^+}$ is an S_1-feasible path and $\mathbf{x}^+ \in S_1$.

Exiting S_1

By Lemma 3.7, $\gamma(t)$ cannot intersect itself and therefore cannot cross $\gamma_1(t)$. Consequently, there are only two ways of exiting S_1:

1. Crossing the level curve. Then f is decreasing on $\gamma(t)$ at the intersection point. Let the next point where $f(\gamma(t))$ starts increasing again be denoted as t^2_{min} and define $\gamma_2(t) = \gamma(t),\ t \in [\hat{t}, t^2_{min}]$. This curve has the same properties as $\gamma_1(t)$:

   (a) $f(\mathbf{x}) \leqslant f(\hat{\mathbf{x}})$ for all $\mathbf{x}$ on $\gamma_2(t)$ and

   (b) $\gamma_2(t)$ only crosses the line $l(\mathbf{x}) = 0$ at $\hat{\mathbf{x}}$ and the level curve $f(\mathbf{x}) = f(\mathbf{x}^2_{min})$ at $\mathbf{x}^2_{min}$, where $\mathbf{x}^2_{min} = \gamma(t^2_{min})$.

   Then S_2 can be defined similarly to S_1 with the new parameters and the same reasoning can be repeated.

2. Cross $l(\mathbf{x}) = 0$. Then $f(\gamma(t)) \leqslant f(\gamma(\hat{t})) \ \forall t \in [\hat{t}, t^1]$.

□

Lemma 3.12. *Consider a point $\hat{\mathbf{x}}$ satisfying the conditions of Lemma 3.11 with $l(\mathbf{x})$ and $\gamma(t)$. Let $\mathbf{x}^1 \in r^i(\hat{\mathbf{x}})$ be the next point where $\gamma(t)$ crosses the line after $\hat{\mathbf{x}}$, then $\mathbf{x}^1$ satisfies the conditions of Lemma 3.11 for $\gamma(t)$ and $-l(\mathbf{x})$.*

Proof. Let t^1 be defined similarly to Lemma 3.11 and $\mathbf{x}^1 = \gamma(t^1)$.

It follows immediately from the definition of $\mathbf{x}^1$ that $l(\mathbf{x}^1) = 0$.

First let us prove that $f(\gamma(t))$ is nonincreasing as a function of t at t^1. Assume the contrary: $f(\gamma(t))$ strictly increases as a function of t at t^1. Then there exists a t^i_{min} such that $\hat{t} < t^i_{min} < t^1$ and $f(\gamma(t))$ is monotone on the $[t^i_{min}, t^1]$ interval.

Then there exists a set S_i and, as proved in the previous lemma, if $\mathbf{x} \in r^i(\hat{\mathbf{x}})$ then $\mathbf{x} \notin S_i$. Then $\gamma(t)$ has to exit S_i at some $t < t^1$. There are two possibilities:

1. $\gamma(t)$ crosses $r^d(\hat{\mathbf{x}})$. This contradicts with $\gamma(t^1) \in r^i(\hat{\mathbf{x}})$,

2. $\gamma(t)$ crosses the level curve. Then $f(\gamma(t))$ decreases somewhere between t^1_{min} and t^1. This contradicts with $f(\gamma(t))$ being monotonic on $[t^i_{min}, t^1]$.

This proves that $f(\gamma(t))$ is nonincreasing at t^1.

Now we shall show that x^1 is a local maximiser of f in $F \cap l(\mathbf{x}) \geqslant 0$.

By Lemma 3.5, $f(\gamma(t))$ being nonincreasing at t^1 implies that:

$$\nabla f(\mathbf{x}^1) \times \nabla g_{i(t^1)}(\mathbf{x}^1) \geqslant 0. \tag{3.6}$$

Since $\gamma(t)$ crosses the line from the $l(\mathbf{x}) \leqslant 0$ half-space into the $l(\mathbf{x}) \geqslant 0$ half-space at $\mathbf{x}^1$, $l(\gamma(t))$ is increasing at t^1 and thus, by Lemma 3.5, we have that $\nabla l \times \nabla g_{i(t^1)}(\mathbf{x}^1) < 0$ or, equivalently:

$$(-\nabla l) \times \nabla g_{i(t^1)}(\mathbf{x}^1) > 0. \tag{3.7}$$

Finally, by Lemma 3.11, $f(\mathbf{x}^1) \leqslant f(\hat{\mathbf{x}})$ and thus $\mathbf{x}^1$ belongs to the part of ray $r^i(\hat{\mathbf{x}})$ where f is decreasing. Since the assumptions on l imply that the direction which $r^i(\hat{\mathbf{x}})$ points at is the positive direction of moving along the line, by Lemma 3.5 we have that

$$\nabla f(\mathbf{x}^1) \times (-\nabla l) \leqslant 0. \tag{3.8}$$

By Lemma 3.6, these inequalities imply that $\mathbf{x}^1$ is a KKT point in $F \cap \{l(\mathbf{x}) \geqslant 0\}$. Thus the conditions of Lemma 3.11 are satisfied at $\mathbf{x}^1$ for $F \cap \{l(\mathbf{x}) \geqslant 0\}$.

□

3.7 Kuhn-Tucker Invexity of Boundary-Invex Problems

3.7.1 Sequence of crossing points

Consider a point $\mathbf{x}^* \in \partial F$ which is a local maximum of (NLP_2) and a linear function l such that f is not constant on $l(\mathbf{x}) = 0$. Let $\gamma(0) = \mathbf{x}^*$.

Given two parameter values r, s, let $\hat{\gamma}(r, s)$ denote the segment of the $\gamma(t)$ curve with $t \in [r, s]$.

Let $\mathbf{x}^i$ be the i^{th} point where $\gamma(t)$ crosses $l(\mathbf{x}) = 0$ and let t^i be a parameter value such that $\mathbf{x}^i = \gamma(t^i)$. Since $\gamma(t)$ is a closed curve, $\mathbf{x}^i$ exists for each $i \in \mathbb{N}$ if at least one crossing point exists.

The numbering of the crossing points will be chosen so that the even indices will correspond to $\gamma(t)$ crossing the line $l(\mathbf{x}) = 0$ from $l(\mathbf{x}) > 0$ into $l(\mathbf{x}) < 0$, and the odd indices will correspond to the opposite direction of crossing.

Lemma 3.13. *Consider a crossing point $\mathbf{x}^i$, $i \in 2\mathbb{N}$. If $\nabla l \times \nabla f(\mathbf{x}^i) > 0$, then $\mathbf{x}^i$ satisfies the assumptions of Lemma 3.11 for either $l(\mathbf{x})$ and $\gamma(t)$ or for $-l(\mathbf{x})$ and $\gamma^r(t)$.*

Proof. Since $\gamma(t)$ crosses the line from $l(\mathbf{x}) > 0$ into $l(\mathbf{x}) < 0$ at $\mathbf{x}^i$, we have that $\nabla l \times \nabla g_{i(t^i)}(\mathbf{x}^i) > 0$.

By Lemma 3.6, $\mathbf{x}^i$ is a KKT point in one of the following sets:

1. $F \cap \{l(\mathbf{x}) \leqslant 0\}$ if $\nabla f \times \nabla g_{i(t^i)} \geqslant 0$. Given the lemma's assumptions, the latter inequality implies that Lemma 3.11 is satisfied at $\mathbf{x}^i$ for $\gamma(t)$ and $l(\mathbf{x})$ (see the beginning of Subsection 3.6.2).

2. $F \cap \{-l(\mathbf{x}) \leqslant 0\}$ if $\nabla f \times \nabla g_{i(t^i)} \leqslant 0$. In this case Lemma 3.11 is satisfied at $\mathbf{x}^i$ for $\gamma^r(t)$ and $-l(\mathbf{x})$.

□

Let $S(\overline{AB}, \overline{BC}, \dots) \subset F$ denote a set with the boundary comprised of some sections of ∂F and segments $\overline{AB}$, $\overline{BC}$, $\dots$ on the line $l(\mathbf{x} = 0)$.

Definition 3.16. *$S(\overline{AB}, \overline{BC}, \dots) \subset F$ is a safe set if $f(\mathbf{x}) \leqslant f(\mathbf{x}^*)$ $\forall \mathbf{x} \in S$.*

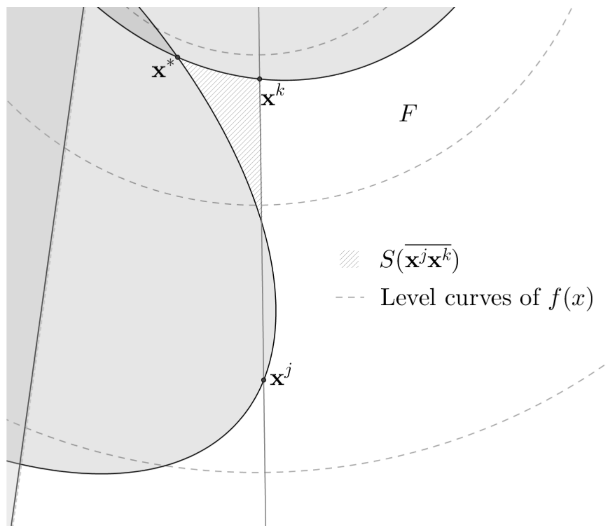


Figure 3.5: Points $\mathbf{x}^*$, $\mathbf{x}^j$, $\mathbf{x}^k$ and set $S(\overline{\mathbf{x}^j\mathbf{x}^k})$ satisfying the conditions of Theorem 3.10

Theorem 3.10. *Consider points $\mathbf{x}^j$, $\mathbf{x}^k \in F$ such that:*

$\mathbf{x}^k$, $k \in 2\mathbb{N}$, satisfies the assumptions of Lemma 3.11 for γ^r and $-l$, $f(\mathbf{x}^k) \leqslant f(\mathbf{x}^*)$;

$\mathbf{x}^j \in r^d(\mathbf{x}^k)$, $j \in 2\mathbb{N}$, satisfies the assumptions of Lemma 3.11 for γ and l;

$f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [0, t^j]$;

if $\mathbf{x}^j \neq \mathbf{x}^k$ and $\gamma(t)$ crosses $\overline{\mathbf{x}^j\mathbf{x}^k}$ from $l(\mathbf{x}) > 0$ into $l(\mathbf{x}) < 0$, it enters a safe set $S(\overline{\mathbf{x}^j\mathbf{x}^k})$ with the boundary consisting of $\overline{\mathbf{x}^j\mathbf{x}^k}$ and $\hat{\gamma}(t^k, t^{j-1})$.

Then x^* is the global optimum of (NLP$_2$)*.*

Proof. The conditions on $\mathbf{x}^k$ imply that $\nabla f(\mathbf{x}^k) \times \nabla l < 0$. By Lemma 3.10, f is monotonically decreasing on the whole ray $r^d(\mathbf{x}^k)$ and thus $\nabla f(\mathbf{x}) \times \nabla l < 0$ $\forall \mathbf{x} \in r^d(\mathbf{x}^k)$. Then points $\mathbf{x}^i \in r^d(\mathbf{x}^k)$, $i \in 2\mathbb{N}$, satisfy conditions of Lemma 3.13.

Let us consider the following cases:

1. $\mathbf{x}^{j+1} \in r^d(\mathbf{x}^k)$.

   Let $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ be the set with the boundary composed of $\hat{\gamma}(t^j, t^{j+1})$ and the segment $\overline{\mathbf{x}^j\mathbf{x}^{j+1}}$. By Lemma 3.11, $f(\mathbf{x}) \leqslant f(\mathbf{x}^j)$ $\forall \mathbf{x} \in \hat{\gamma}(t^j, t^{j+1})$. Since the segment $\overline{\mathbf{x}^j\mathbf{x}^{j+1}}$ is part of the $r^d(\mathbf{x}^j)$ ray, then by Lemma 3.10, f is decreasing on this segment from x^j in the direction of x^{j+1} and thus $f(\mathbf{x}) \leqslant f(\mathbf{x}^j)$ $\forall \mathbf{x} \in \overline{\mathbf{x}^j\mathbf{x}^{j+1}}$. Since $\mathbf{x}^j$ satisfies the conditions of Lemma 3.11, it is a local maximum in $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$. Then, by Lemma 3.1, $f(\mathbf{x}) \leqslant f(\mathbf{x}^j) \leqslant f(\mathbf{x}^*)$ $\forall \mathbf{x} \in S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$. Thus $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ is a safe set.

By Lemma 3.7, $\gamma(t)$ cannot exit $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ by crossing itself. Then the only way to exit $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ is to cross the $\overline{\mathbf{x}_j\mathbf{x}_{j+1}}$ line segment again.

Let $S(\overline{\mathbf{x}^j\mathbf{x}^k}, \overline{\mathbf{x}^j\mathbf{x}^{j+1}}) = S(\overline{\mathbf{x}^j\mathbf{x}^k}) \cup S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$. Since it is a union of two safe sets, $S(\overline{\mathbf{x}^j\mathbf{x}^k}, \overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ is a safe set.

If $\mathbf{x}^{j+2} = \mathbf{x}^k$, then, by Lemma 3.11 applied to x^k, $-l$ and γ^r, $f(\mathbf{x}) \leqslant f(\mathbf{x}^k) \leqslant f(\mathbf{x}^*)$ $\forall \mathbf{x} \in \hat{\gamma}(t^{j+1}, t^{j+2})$. Since the conditions of the theorem imply that $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [k, T] \cup [0, j]$, we have that $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [0, T]$.

We will consider the following cases that depend on the position of $\mathbf{x}^{j+2}$ on $l(\mathbf{x}) = 0$:

(a) $\mathbf{x}^{j+2} \in \overline{\mathbf{x}^k\mathbf{x}^{j+1}}$.

$\gamma(t)$ enters $S(\overline{\mathbf{x}^j\mathbf{x}^k}, \overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ at $\mathbf{x}^{j+2}$. Since it is a safe set, f cannot reach values larger than $f(\mathbf{x}^*)$ unless there exists $\mathbf{x}^p \in \overline{\mathbf{x}^k\mathbf{x}^{j+1}}$, $p \in 2\mathbb{N}{+}1$ such that γ exits $S(\overline{\mathbf{x}^j\mathbf{x}^k}, \overline{\mathbf{x}^j\mathbf{x}^{j+1}})$ at $\mathbf{x}^p$. Repeat case (1) with $\mathbf{x}^p$ instead of $\mathbf{x}^{j+1}$.

(b) $\mathbf{x}^{j+2} \in r^d(\mathbf{x}^{j+1})$.

$j{+}2 \in 2\mathbb{N}$. Since $\mathbf{x}^{j+2} \in r^d(\mathbf{x}^{j+1}) \in r^d(\mathbf{x}^k)$ and, by Lemma 3.10, a concave function is always decreasing in the direction of local decrease from a given point, the monotonicity of f on $l(\mathbf{x}) = 0$ at $\mathbf{x}^{j+2}$ is similar to that at $\mathbf{x}^k$. This implies that $\mathrm{sign}(\nabla l \times \nabla f(\mathbf{x}^{j+2})) = \mathrm{sign}(\nabla l \times \nabla f(\mathbf{x}^k)) > 0$. Then, by Lemma 3.13, one of the following is true at $\mathbf{x}^{j+2}$:

i. $f(\gamma(t))$ is increasing at t^{j+2}. Then at this point Lemma 3.11 can be applied for γ^r and $-l$ to show that $f(\gamma(t)) \leqslant f(\mathbf{x}^{j+2})$ $\forall t \in (t^{j+1}, t^{j+2})$. But by Lemma 3.10, f is nonincreasing on $r^d(\mathbf{x}^k)$ and $f(\mathbf{x}^{j+2}) < f(\mathbf{x}^{j+1})$. This contradicts with $f(\mathbf{x}^{j+2}) \leqslant f(\mathbf{x}^{j+1})$.

ii. $f(\gamma(t))$ is decreasing at t^{j+2}. Then Lemma 3.11 is satisfied at $\mathbf{x}^{j+2}$ for γ and l. Then $\mathbf{x}^{j+2}, \mathbf{x}^k$ and the $l(\mathbf{x}) = 0$ line satisfy the conditions of this theorem and the reasoning can be repeated from the start.

(c) $\mathbf{x}^{j+2} \in r^i(\mathbf{x}^k)$.

Let $S(\overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}})$ be the set with the boundary composed of $\overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}}$ and $\gamma(t^{j+1}, t^{j+2})$. Let $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^k}) = S(\overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}}) \cup S(\overline{\mathbf{x}^k\mathbf{x}^j}) \cup S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$.

At $\mathbf{x}^{j+2}$ $\gamma(t)$ leaves $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^k})$. But $\mathbf{x}^k$ belongs to the boundary of $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^k})$ and $\gamma(t)$ approaches $\mathbf{x}^k$ from the interior of this set. This implies that at some point $\gamma(t)$ enters $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^k})$. Let $\mathbf{x}^m$ denote the last such point on $\gamma(t)$ before x^k. Then the next crossing point $\mathbf{x}^{m+1}$ can only belong to $\overline{\mathbf{x}^k\mathbf{x}^{j+1}}$.

Properties of t^m

Consider the point $\mathbf{x}^{k-1}$. If $\mathbf{x}^{k-1} = \mathbf{x}^m$, then, by Lemma 3.12 applied to t^k and t^m, $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [t^k, t^m]$ and $\mathbf{x}^m$ satisfies the conditions of Lemma 3.11 for l and γ^r.

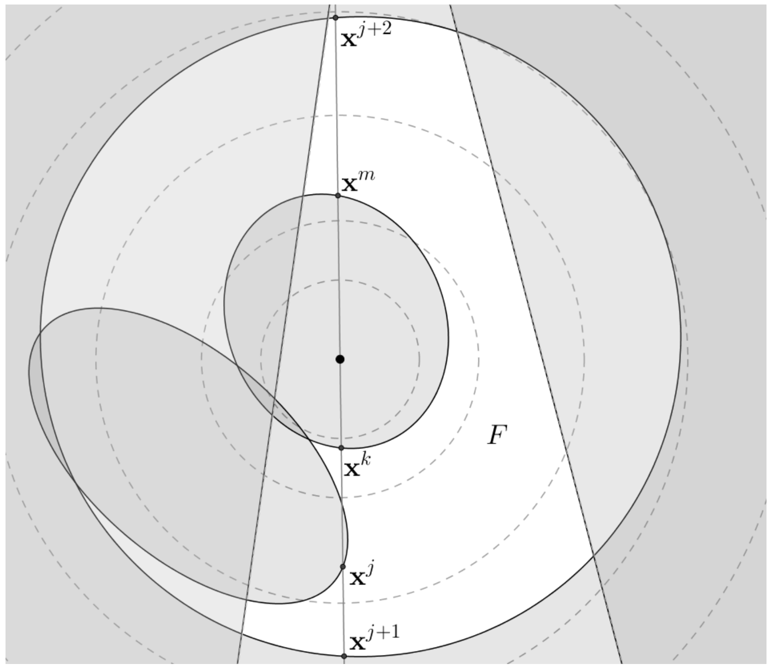


Figure 3.6: Case (1c): $\mathbf{x}^{j+1} \in r^d(\mathbf{x}^k)$

Now suppose that $\mathbf{x}^{k-1} \neq \mathbf{x}^m$. The definition of $\mathbf{x}^m$ implies that $\mathbf{x}^{k-1} \in \overline{\mathbf{x}^k\mathbf{x}^{j+1}}$. The points following $\mathbf{x}^{k-1}$ on $\gamma^r(t)$ belong to one of the sets $S(\overline{\mathbf{x}^k\mathbf{x}^j})$, $S(\overline{\mathbf{x}^j\mathbf{x}^{j+1}})$. Thus $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [k-1, k]$ and $\mathbf{x}^{k-2}$ exists and belongs to $\overline{x^k x^{j+1}}$. Consider the following cases:

i. $\mathbf{x}^{k-2} \in \overline{\mathbf{x}^k\mathbf{x}^{k-1}}$. Then $\gamma^r(t)$ enters the set $S(\overline{\mathbf{x}^k\mathbf{x}^{k-1}})$ and $\mathbf{x}^{k-3} \neq x^m$. Repeat case (1ci) or (1cii) with $\mathbf{x}^{k-3}$ instead of $\mathbf{x}^{k-1}$.

ii. $\mathbf{x}^{k-2} \in \overline{\mathbf{x}^{k-1}\mathbf{x}^{j+1}}$. Then, similarly to case (1b), $\mathbf{x}^{k-2}$ satisfies the conditions of Lemma 3.11 for $-l$ and γ^r. Thus $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [t^{k-2}, t^{k-3}]$. If $\mathbf{x}^{k-3} = \mathbf{x}^m$, by Lemma 3.12 $\mathbf{x}^m$ satisfies the conditions of Lemma 3.11 for l and γ. Otherwise repeat (1ci) or (1cii) with $\mathbf{x}^{k-2}$ instead of $\mathbf{x}^k$ and $\mathbf{x}^{k-3}$ instead of $\mathbf{x}^{k-1}$.

We have proven that $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [t^k, t^m]$ and $\mathbf{x}^m$ satisfies the conditions of Lemma 3.11 for l and γ^r.

Starting a new iteration

Consider the set $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ that contains the section of the $\gamma(t)$ curve from t^m to t^{j+2}. If this set is disconnected, then there exist points $\mathbf{x} \in S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ that cannot be connected to the segment $\overline{\mathbf{x}^{j+2}\mathbf{x}^m}$ by a continuous path that belongs to this set. But since every feasible path from $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ to $F\backslash S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ crosses $\overline{\mathbf{x}^{j+2}\mathbf{x}^m}$, this implies that there is no feasible path from $\mathbf{x}$ to points in $F\backslash S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ and thus F is disconnected. This contradicts with the theorem assumptions. Hence

$S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$ is a connected set.

We have shown that $f(\mathbf{x}) \leqslant f(\mathbf{x}^*)$ for all $\mathbf{x}$ on this curve. $\mathbf{x}^*$ is a local maximum in $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$. Then $f(\mathbf{x}) \leqslant f(\mathbf{x}^*)$ $\forall \mathbf{x} \in S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$.

Case (1) of this theorem can be repeated with $\mathbf{x}^{j+2}$, $S(\overline{\mathbf{x}^{j+2}\mathbf{x}^m})$, $-l$ and $\mathbf{x}^m$ instead of $\mathbf{x}^{j+1}$, $S(\overline{\mathbf{x}^j\mathbf{x}^k}, \overline{\mathbf{x}^j\mathbf{x}^{j+1}})$, l and $\mathbf{x}^k$.

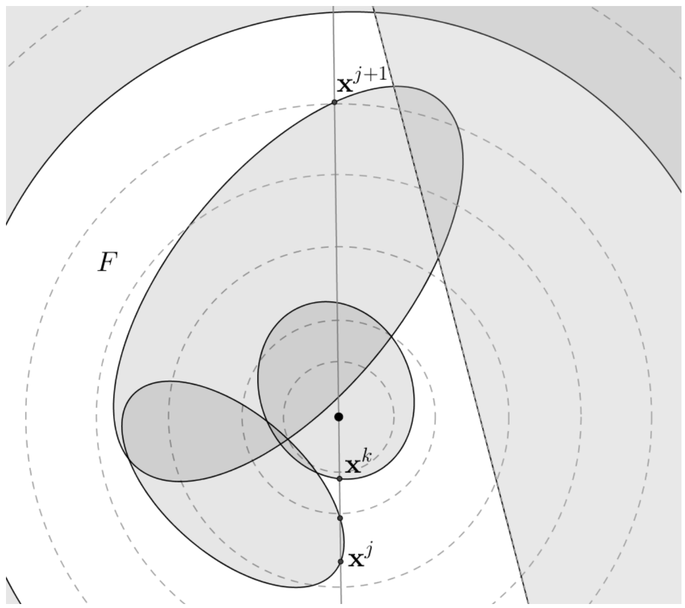


Figure 3.7: Case 2: $\mathbf{x}^{j+1} \in r^i(\mathbf{x}^k)$

2. $\mathbf{x}^{j+1} \in r^i(\mathbf{x}^k)$. By Lemma 3.12, $f(\gamma(t))$ is decreasing at t^{j+1} and $\mathbf{x}^{j+1}$ satisfies the conditions of Lemma 3.11. Then $f(\mathbf{x}) \leqslant f(\mathbf{x}^{j+1})$ until the next crossing point $\mathbf{x}^{j+2}$.

   (a) $\mathbf{x}^{j+2} \in \overline{\mathbf{x}^k\mathbf{x}^{j+1}}$.

   The assumptions of this theorem imply that $f(\mathbf{x}^k) > f(\mathbf{x}_j) \geqslant f(\mathbf{x}^{j+1})$. This means that $\mathbf{x}^k$ belongs to the increasing section of the ray $r^i(\mathbf{x}^{j+1})$ and $f(\mathbf{x}) > f(\mathbf{x}^{j+1})$ $\forall \mathbf{x} \in \overline{\mathbf{x}^k\mathbf{x}^{j+1}}$. Then $f(\mathbf{x}^{j+2}) > f(\mathbf{x}^{j+1})$. Contradiction with $f(\mathbf{x}^{j+2}) \leqslant f(\mathbf{x}^{j+1})$.

   (b) $\mathbf{x}^{j+2} \in r^d(\mathbf{x}^k)$.

   By applying Lemma 3.12 to $\mathbf{x}^{j+2}$ we can show that this point satisfies the conditions of Lemma 3.11 for γ and l. Then $\mathbf{x}^{j+2}$ has the same properties as $\mathbf{x}^j$. Repeat everything with same $\mathbf{x}^k$ and $\mathbf{x}^{j+2}$ instead of $\mathbf{x}^j$.

   (c) $\mathbf{x}^{j+2} \in r^d(\mathbf{x}^{j+1})$.

      i. $\mathbf{x}^{j+3} \in r^d(\mathbf{x}^{j+2})$. Repeat (2) with $\mathbf{x}^{j+3}$ instead of $\mathbf{x}^{j+1}$.

ii. $\mathbf{x}^{j+3} \in \overline{\mathbf{x}^k \mathbf{x}^{j+2}}$. From $\mathbf{x}^{j+2}$ $\gamma(t)$ cannot reach the line segment $\overline{\mathbf{x}^k \mathbf{x}_{j+1}}$ without crossing $\hat{\gamma}(t^j, t^{j+1})$. Then $\mathbf{x}^{j+2} \in \overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}}$ and $\gamma(t)$ enters a safe set $S(\overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}})$. Then $\mathbf{x}^{j+4} \in \overline{\mathbf{x}^{j+1}\mathbf{x}^{j+2}}$. Repeat (2c) with $\mathbf{x}^{j+4}$ instead of $\mathbf{x}^{j+2}$.

iii. $\mathbf{x}^{j+3} \in r^d(\mathbf{x}^k)$. Repeat (1) with $\mathbf{x}^{j+3}$ instead of $\mathbf{x}^{j+1}$.

We have proven that $f(\gamma(t)) \leqslant f(\mathbf{x}^*)$ $\forall t \in [0, T]$. By Lemma 3.1, together with the fact that $\mathbf{x}^*$ is a local maximum this implies that $\mathbf{x}^*$ is the global maximum of (NLP$_2$).

□

3.7.2 The main theorem

Theorem 3.11. *If* (NLP$_2$) *is boundary-invex, then it is KT-invex.*

Proof. If no KKT points exist for (NLP$_2$), then it is KT-invex.

Let $\mathbf{x}^*$ be a KKT point of (NLP$_2$). If $\mathbf{x}^*$ lies in the interior of F, then, by concavity of f, it is the global unconstrained maximum of f and thus the global maximum for (NLP$_2$).

Now suppose that $\mathbf{x}^* \in \partial F$. Let $\gamma(0) = \gamma(T) = \mathbf{x}^*$ in the parametrisation of ∂F. By Lemma 3.1, it is enough to consider only the values on the boundary. We need to prove that there exists a line $l(\mathbf{x}) = 0$ such that the conditions of Theorem 3.10 are satisfied for the point $\mathbf{x}^* = \mathbf{x}_j = \mathbf{x}_k$.

By Lemma 3.2, either one or two constraints can be nonredundant around $\mathbf{x}^*$. Consider the case where two constraints are nonredundant around $\mathbf{x}^*$ and let these constraints be denoted g_1, g_2.

If $\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) = 0$, then $\nabla g_1(\mathbf{x}^*) = c\nabla g_2(\mathbf{x}^*)$, and LICQ is violated. Now suppose that $\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) \neq 0$. Since $\mathbf{u} \times \mathbf{v} = -(\mathbf{v} \times \mathbf{u})$, we can assume w.l.o.g. that $\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) < 0$. Then, by Lemma 3.6, the following holds:

$$\nabla f(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) \geqslant 0, \tag{3.9}$$
$$\nabla f(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) \leqslant 0 \tag{3.10}$$

and, since $\mathbf{x}^*$ satisfies LICQ, at least one of the above inequalities is strict.

Consider a linear function l such that $l(\mathbf{x}^*) = 0$ and $\nabla l = -\nabla g_1(\mathbf{x}^*) + \nabla g_2(\mathbf{x}^*)$. By (3.9), (3.10) we have:

$$\nabla f(\mathbf{x}^*) \times \nabla l = -\nabla f(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) + \nabla f(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) < 0 \tag{3.11}$$

and

$$\begin{aligned}\nabla l \times \nabla g_1(\mathbf{x}^*) = -\nabla g_1(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) + \nabla g_2(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) = \\ \nabla g_2(\mathbf{x}^*) \times \nabla g_1(\mathbf{x}^*) > 0,\end{aligned} \tag{3.12a}$$
$$\begin{aligned}\nabla l \times \nabla g_2(\mathbf{x}^*) = -\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) + \nabla g_2(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) = \\ -\nabla g_1(\mathbf{x}^*) \times \nabla g_2(\mathbf{x}^*) > 0.\end{aligned} \tag{3.12b}$$

By Lemma 3.6, the inequality (3.11) together with inequalities (3.10) and (3.12b) (resp. (3.9) and(3.12a)) imply that $\mathbf{x}^*$ satisfies the conditions of Lemma 3.11 for γ and l (γ^r and $-l$ resp.). $\nabla f(\mathbf{x}^*) \times \nabla l < 0$ also implies that f is nonconstant on $l(\mathbf{x}) = 0$. Theorem 3.10 can then be applied with $\mathbf{x}^k = \mathbf{x}^j = \mathbf{x}^*$ to show that $\mathbf{x}^*$ is the global maximum of (NLP_2).

If one constraint g_1 is nonredundant around $\mathbf{x}^*$, choose l such that $l(\mathbf{x}^*) = 0$ and $\nabla l = (\frac{\partial g_1}{\partial y}, -\frac{\partial g_1}{\partial x})$. Since $\mathbf{x}^*$ is a KKT point of (NLP_2), it satisfies:

$$\nabla f(\mathbf{x}^*) = c\nabla g_1(\mathbf{x}^*),\ c > 0.$$

By our choice of the function l, the following holds:

$$\nabla f(\mathbf{x}^*) \times \nabla l < 0,$$

$$\nabla l \times \nabla g_1(\mathbf{x}^*) > 0.$$

Similarly to the case of two constraints, Lemma 3.2 and Theorem 3.10 imply that $\mathbf{x}^*$ is the global maximum of (NLP_2).

We have shown that any KKT point of (NLP_2) is a global optimum. Thus (NLP_2) is KT-invex.

□

3.8 Application: KT-Invexity of AC-OPF

The AC Optimal Power Flow problem given by Model 2.1 in Chapter 2 is a nonconvex nonlinear optimisation problem, and in general it is NP-Hard [186, 124]. In the following section we look at a family of AC-OPF problems with two degrees of freedom and show that they are boundary-invex under mild assumptions on the variables' bounds and signs. Namely, we will enforce that

$$-\frac{\pi}{6} \leqslant \boldsymbol{\theta}^l_{ij} < \boldsymbol{\theta}^u_{ij} \leqslant \frac{\pi}{6},\ 0.95 \leqslant \boldsymbol{v}^{\boldsymbol{l}}_i < \boldsymbol{v}^{\boldsymbol{u}}_i \leqslant 1.05,\ \boldsymbol{g}_{ij} \geqslant 0 \text{ and } \boldsymbol{b}_{ij} < 0.$$

3.8.1 Boundary-invex AC-OPF

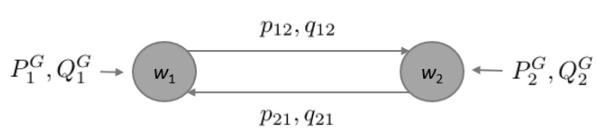


Figure 3.8: 1-line network

Consider a 2-bus network with one line and two generators as depicted in Figure 3.8. Here we will use the w-formulation of AC-OPF as presented in Subsection 2.4.1:

Model 3.1: The w-formulation of AC-OPF for a 1-line network

variables

$p_{12},\ q_{12},\ p_{21},\ q_{21}$

$w_{12}^R \in [(\boldsymbol{w}_{12}^R)^l,\ (\boldsymbol{w}_{12}^R)^u],\ w_{12}^I \in [(\boldsymbol{w}_{12}^I)^l,\ (\boldsymbol{w}_{12}^I)^u]$

$w_i \in [(\boldsymbol{v}_i^l)^2,\ (\boldsymbol{v}_i^u)^2],\ i = 1, 2$

$p_i^g \in [\boldsymbol{p}_i^{gl},\ \boldsymbol{p}_i^{gu}],\ q_i^g \in [\boldsymbol{q}_i^{gl},\ \boldsymbol{q}_i^{gu}],\ i = 1, 2$

objective:

$$\min \sum_{i=1,2} \left(\boldsymbol{c}_{2i}(p_i^g)^2 + \boldsymbol{c}_{1i}p_i^g + \boldsymbol{c}_{0i}\right)$$

subject to:

$$p_i^g - \boldsymbol{p}_i^d = p_{ij},\ \ i = 1, 2 \tag{3.13a}$$

$$q_i^g - \boldsymbol{q}_i^d = q_{ij},\ \ i = 1, 2 \tag{3.13b}$$

$$p_{12} = \mathbf{g}_{12}w_1 - \mathbf{g}_{12}w_{12}^R - \mathbf{b}_{12}w_{12}^I \tag{3.13c}$$

$$q_{12} = -\mathbf{b}_{12}w_1 + \mathbf{b}_{12}w_{12}^R - \mathbf{g}_{12}w_{12}^I \tag{3.13d}$$

$$p_{21} = \mathbf{g}_{12}w_2 - \mathbf{g}_{12}w_{12}^R + \mathbf{b}_{12}w_{12}^I \tag{3.13e}$$

$$q_{21} = -\mathbf{b}_{12}w_2 + \mathbf{b}_{12}w_{12}^R + \mathbf{g}_{12}w_{12}^I \tag{3.13f}$$

$$p_{12}^2 + q_{12}^2 \leqslant \boldsymbol{s}_{12}^{\boldsymbol{u}} \tag{3.13g}$$

$$w_{12}^R \tan(\boldsymbol{\theta}_{12}^l) \leqslant w_{12}^I \leqslant w_{12}^R \tan(\boldsymbol{\theta}_{12}^u) \tag{3.13h}$$

$$(w_{12}^R)^2 + (w_{12}^I)^2 = w_1 w_2 \tag{3.13i}$$

The linear part and the thermal limit constraints are here the same as those presented in Subsection 2.4.1. The only difference is in the representation of the remaining nonlinear nonconvex relations. Instead of imposing conditions on the matrix W, we include equation (3.13i) into Model 3.1 in order to ensure equivalence to the AC-OPF model 2.1.

We assume the voltage magnitude to be fixed at node 1. For clarity purposes we will adopt the following notations: $\boldsymbol{w} = \boldsymbol{w}_1$, $w^R = w_{12}^R$, $w^I = w_{12}^I$ and $\boldsymbol{s}^u = \boldsymbol{s}_{12}^u$. By substituting p_{ij} and q_{ij} with their expressions as in equations (3.13c)-(3.13f) and using equation (3.13i) to express w_2 through the other w-variables:

$$w_2 = \frac{(w^R)^2 + (w^I)^2}{\boldsymbol{w}},$$

we obtain the following projected formulation:

Model 3.2: Projected AC-OPF for 1-line networks

variables:

$w^R,\ w^I$

objective:

$$\min \ \Big(\boldsymbol{c}_1(\boldsymbol{gw} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + \boldsymbol{c}_2(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I)\Big) \tag{3.14a}$$

subject to:

$$(\boldsymbol{gw} - \boldsymbol{g}w^R - \boldsymbol{b}w^I)^2 + (-\boldsymbol{bw} + \boldsymbol{b}w^R - \boldsymbol{g}w^I)^2 \leqslant \boldsymbol{s}^u \tag{3.14b}$$

$$(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I)^2 + (-\frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{b}w^R + \boldsymbol{g}w^I)^2 \leqslant \boldsymbol{s}^u \tag{3.14c}$$

$$(\boldsymbol{p}_1^g)^l - \boldsymbol{p}_1^d \leqslant \boldsymbol{gw} - \boldsymbol{g}w^R - \boldsymbol{b}w^I \leqslant (\boldsymbol{p}_1^g)^u - \boldsymbol{p}_1^d \tag{3.14d}$$

$$(\boldsymbol{q}_1^g)^l - \boldsymbol{q}_1^d \leqslant -\boldsymbol{bw} + \boldsymbol{b}w^R - \boldsymbol{g}w^I \leqslant (\boldsymbol{q}_1^g)^u - \boldsymbol{q}_1^d \tag{3.14e}$$

$$(\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d \leqslant \frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I \leqslant (\boldsymbol{p}_2^g)^u - \boldsymbol{p}_2^d \tag{3.14f}$$

$$(\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d \leqslant -\frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{b}w^R + \boldsymbol{g}w^I \leqslant (\boldsymbol{q}_2^g)^u - \boldsymbol{q}_2^d \tag{3.14g}$$

$$(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \leqslant \frac{(w^R)^2 + (w^I)^2}{\boldsymbol{w}} \leqslant (\boldsymbol{v}_2^{\boldsymbol{u}})^2 \tag{3.14h}$$

$$w^R \tan(\boldsymbol{\theta}^l) \leqslant w^I \leqslant w^R \tan(\boldsymbol{\theta}^u) \tag{3.14i}$$

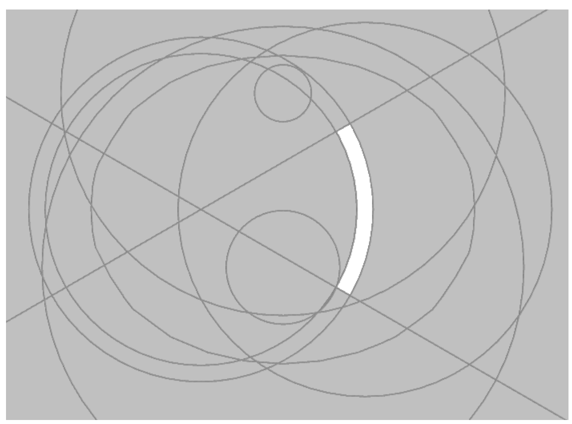


Figure 3.9: Feasible set of Model 2 (feasible region in white)

Model 3.2 has four nonconvex constraints, namely, (3.14c) and the lower bounds in (3.14f)-(3.14h). The feasible set of this problem is shown in Figure 3.9.

3.8.2 Invexity proof for 1-line AC-OPF

Minimal feasible w^R

Lemma 3.14. *If $w^R \leqslant 0.77\boldsymbol{w}$, then (w^R, w^I) is infeasible for Model 3.2.*

Proof. Consider the lower bound on the squared voltage magnitude at bus 2 (left hand side of constraint (3.14h)) and voltage angle bounds (3.14i). No feasible points exist where:

$$w^R \tan(\boldsymbol{\theta}^l) \leqslant w^I \leqslant w^R \tan(\boldsymbol{\theta}^u) \implies (w^R)^2 + (w^I)^2 < (\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w}.$$

If $w^R \geqslant (\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w}$, the latter is always false. Suppose that $w^R < (\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w}$. Consider the $w^I \geqslant 0$ half-space. The lower angle bound is redundant here, and the remaining two inequalities can be written as:

$$w^I \leqslant w^R \tan(\boldsymbol{\theta}^u),$$
$$w^I < \sqrt{(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w} - (w^R)^2}.$$

The implication holds if the second inequality is dominated by the first:

$$w^R \tan(\boldsymbol{\theta}^u) < \sqrt{(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w} - (w^R)^2} \Leftrightarrow$$
$$(w^R)^2 \tan^2(\boldsymbol{\theta}^u) < (\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w} - (w^R)^2.$$

It can be seen that only points with nonnegative w^R can satisfy constraint (3.14i). Then the above is equivalent to:

$$w^R < \sqrt{\frac{(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w}}{\tan^2(\boldsymbol{\theta}^u) + 1}}.$$

Since $\boldsymbol{\theta}^u \leqslant \frac{\pi}{6}$ and $(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \geqslant (0.95)^2$, we have that:

$$\sqrt{\frac{(\boldsymbol{v}_2^{\boldsymbol{l}})^2 \boldsymbol{w}}{\tan^2(\boldsymbol{\theta}^u) + 1}} \geqslant \sqrt{\frac{(0.95)^2 \boldsymbol{w}}{\frac{1}{3} + 1}} \geqslant 0.82\sqrt{\boldsymbol{w}}.$$

All $w^R < 0.82\sqrt{\boldsymbol{w}}$ are guaranteed to be infeasible. Since $\boldsymbol{w} \leqslant 1.1025$, it can be shown that $\sqrt{\boldsymbol{w}} \geqslant 0.95\boldsymbol{w}$ and thus all $w^R < 0.77\boldsymbol{w}$ are infeasible.

□

Lower bound on the squared voltage magnitude at bus 2 Consider constraint (3.14h). Let $g_1(w^R, w^I) = (\boldsymbol{v}^{\boldsymbol{l}})^2 - \frac{(w^R)^2 + (w^I)^2}{\boldsymbol{w}}$.

Lemma 3.15. *KKT points of problem* (NLP_{2i})*,* $i = 1$ *do not violate the boundary-invexity for Model 3.2.*

Proof. (NLP_{2i}) takes the following form for $i = 1$:

$$\max \ \left(c_1(\boldsymbol{g}\boldsymbol{w} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + c_2\left(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I\right)\right)$$
$$s.t. \ (\boldsymbol{v}_{\boldsymbol{2}}^{\boldsymbol{l}})^2 \boldsymbol{w} - (w^R)^2 - (w^I)^2 = 0,$$

which can be rewritten as

$$
\begin{aligned}
&\max \ \Big(\boldsymbol{c}_1(\boldsymbol{g}\boldsymbol{w} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + \boldsymbol{c}_2(\frac{\boldsymbol{g}}{\boldsymbol{w}}(\boldsymbol{v^l})^2\boldsymbol{w} - \boldsymbol{g}w^R + \boldsymbol{b}w^I)\Big) \\
&s.t. \ (\boldsymbol{v_2^l})^2\boldsymbol{w} - (w^R)^2 - (w^I)^2 = 0.
\end{aligned}
$$

The KKT conditions for this problem are:

$$
\begin{aligned}
-\boldsymbol{g}(\boldsymbol{c}_1 + \boldsymbol{c}_2) &= -2\lambda\hat{w}^R, \\
\boldsymbol{b}(\boldsymbol{c}_2 - \boldsymbol{c}_1) &= -2\lambda\hat{w}^I, \\
((\boldsymbol{v_2^l})^2\boldsymbol{w} - (\hat{w}^R)^2 - (\hat{w}^I)^2 &= 0.
\end{aligned}
$$

The solution of this system can violate boundary-invexity only if $\lambda < 0$. It can be seen from the first equation that $\hat{w}^R < 0$ if $\lambda < 0$. But since by Lemma 3.14 all points (w^R, w^I) such that $w^R < 0.77\boldsymbol{w}$ are infeasible, $(\hat{w}^R, \hat{w}^I)$ is infeasible if $\lambda < 0$.

□

p_{21} **lower bound** Consider constraint (3.14f). Let $g_2 = (\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d - \frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{g}w^R - \boldsymbol{b}w^I$.

Lemma 3.16. *KKT points of problem* (NLP$_{2i}$)*, $i = 2$ do not violate the boundary-invexity for Model 3.2.*

Proof. (NLP$_{2i}$) takes the following form for $i = 2$:

$$
\begin{aligned}
&\max \ \Big(\boldsymbol{c}_1(\boldsymbol{g}\boldsymbol{w} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + \boldsymbol{c}_2(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I)\Big) \\
&s.t. \ (\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d - \frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{g}w^R - \boldsymbol{b}w^I = 0,
\end{aligned}
$$

which can be rewritten as

$$
\begin{aligned}
&\max \ \big(\boldsymbol{c}_1(\boldsymbol{g}\boldsymbol{w} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + \boldsymbol{c}_2((\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d)\big) \\
&s.t. \ (\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d - \frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{g}w^R - \boldsymbol{b}w^I = 0.
\end{aligned}
$$

The KKT conditions for this problem are:

$$
\begin{aligned}
-\boldsymbol{c}_1\boldsymbol{g} &= \lambda(-\frac{2\boldsymbol{g}\hat{w}^R}{\boldsymbol{w}} + \boldsymbol{g}), \\
-\boldsymbol{c}_1\boldsymbol{b} &= \lambda(-\frac{2\boldsymbol{g}\hat{w}^I}{\boldsymbol{w}} - \boldsymbol{b}), \\
(\boldsymbol{p}_2^g)^l - \boldsymbol{p}_2^d - \frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) + \boldsymbol{g}w^R - \boldsymbol{b}w^I &= 0
\end{aligned}
$$

and the first equation implies that

$$\hat{w}^R = \frac{\boldsymbol{c}_1 \boldsymbol{w}}{2\lambda} + \frac{\boldsymbol{w}}{2}.$$

The solution of this system can violate boundary-invexity only if $\lambda < 0$. Then $\hat{w}^R < \frac{\boldsymbol{w}}{2}$. But since by Lemma 3.14 all points (w^R, w^I) such that $w^R < 0.77\boldsymbol{w}$ are infeasible, $(\hat{w}^R, \hat{w}^I)$ is infeasible if $\lambda < 0$.

□

q_{21} **lower bound** Consider constraint (3.14g). Let $g_3 = (\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d + \frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{b}w^R - \boldsymbol{g}w^I$.

Lemma 3.17. *KKT points of problem* (NLP$_{2i}$), $i = 3$ *do not violate the boundary-invexity for Model 3.2.*

Proof. (NLP$_{2i}$) takes the following form for $i = 3$:

$$\begin{aligned}
&\max \ \Big(\boldsymbol{c}_1(\boldsymbol{g}\boldsymbol{w} - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + c_2(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{g}w^R + \boldsymbol{b}w^I)\Big) \\
&s.t. \ (\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d + \frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{b}w^R - \boldsymbol{g}w^I = 0,
\end{aligned}$$

which can be rewritten as

$$\begin{aligned}
&\max \ \Big(\boldsymbol{c}_1(\boldsymbol{g}w - \boldsymbol{g}w^R - \boldsymbol{b}w^I) + \frac{\boldsymbol{c}_2}{\boldsymbol{b}}((\boldsymbol{b}^2 + \boldsymbol{g}^2)w^I - \boldsymbol{g}((\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d))\Big) \\
&s.t. \ (\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d + \frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{b}w^R - \boldsymbol{g}w^I = 0.
\end{aligned}$$

The KKT conditions for this problem are:

$$\begin{aligned}
&-\boldsymbol{c}_1\boldsymbol{g} = \lambda(\frac{2\boldsymbol{b}\hat{w}^R}{\boldsymbol{w}} - \boldsymbol{b}), \\
&-\boldsymbol{c}_1\boldsymbol{b} + \frac{\boldsymbol{c}_2(\boldsymbol{b}^2 + \boldsymbol{g}^2)}{\boldsymbol{b}} = \lambda(\frac{2\boldsymbol{b}\hat{w}^I}{\boldsymbol{w}} - \boldsymbol{g}), \\
&(\boldsymbol{q}_2^g)^l - \boldsymbol{q}_2^d + \frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2 + (w^I)^2) - \boldsymbol{b}w^R - \boldsymbol{g}w^I = 0
\end{aligned}$$

and the first equation implies that

$$\hat{w}^R = -\frac{\boldsymbol{c}_1\boldsymbol{g}\boldsymbol{w}}{2\boldsymbol{b}\lambda} + \frac{\boldsymbol{w}}{2}.$$

The solution of this system can violate boundary-invexity only if $\lambda < 0$. Then, given that $g \geqslant 0$ and $b < 0$, we have that $\hat{w}^R < \frac{\boldsymbol{w}}{2}$. But since by Lemma 3.14 all points (w^R, w^I) such that $w^R < 0.77\boldsymbol{w}$ are infeasible, $(\hat{w}^R, \hat{w}^I)$ is infeasible if $\lambda < 0$.

□

We now consider the thermal limit constraint (3.14c).

Lemma 3.18. *If constraint (3.14c) is nonredundant in a given subset, it is locally convex in this subset.*

Proof. Consider the boundary of the set defined by constraint (3.14c). It is given by:

$$
\begin{aligned}
&(\frac{\boldsymbol{g}}{\boldsymbol{w}}((w^R)^2+(w^I)^2)-\boldsymbol{g}w^R+\boldsymbol{b}w^I)^2+(-\frac{\boldsymbol{b}}{\boldsymbol{w}}((w^R)^2+(w^I)^2)+\boldsymbol{b}w^R+\boldsymbol{g}w^I)^2\\
&-\boldsymbol{s}^u=\frac{\boldsymbol{g}^2}{\boldsymbol{w}^2}s^2+\frac{2\boldsymbol{g}}{\boldsymbol{w}}s(\boldsymbol{b}w^I-\boldsymbol{g}w^R)+(\boldsymbol{b}w^I-\boldsymbol{g}w^R)^2+\frac{\boldsymbol{b}^2}{\boldsymbol{w}^2}s^2-\frac{2\boldsymbol{b}}{\boldsymbol{w}}s(\boldsymbol{b}w^R+\boldsymbol{g}w^I)\\
&+(\boldsymbol{b}w^R+\boldsymbol{g}w^I)^2-\boldsymbol{s}^u=\frac{|\boldsymbol{Y}|}{\boldsymbol{w}^2}s^2+\frac{2}{\boldsymbol{w}}s(-\boldsymbol{g}^2w^R-\boldsymbol{b}^2w^R)+(\boldsymbol{b}w^I-\boldsymbol{g}w^R)^2\\
&+(\boldsymbol{b}w^R+\boldsymbol{g}w^I)^2-\boldsymbol{s}^u=\frac{|\boldsymbol{Y}|}{\boldsymbol{w}^2}s^2-\frac{2w^R|\boldsymbol{Y}|}{\boldsymbol{w}}s+|\boldsymbol{Y}|s-\boldsymbol{s}^u=0,
\end{aligned}
$$

where $s=(w^R)^2+(w^I)^2$ and $|\boldsymbol{Y}|=\boldsymbol{g}^2+\boldsymbol{b}^2$. This equation has the following solutions:

$$
\begin{aligned}
s&=\frac{\boldsymbol{w}}{2}\left(2w^R-\boldsymbol{w}-\sqrt{(2w^R-\boldsymbol{w})^2+\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}}\right)\quad\text{and}\\
s&=\frac{\boldsymbol{w}}{2}\left(2w^R-\boldsymbol{w}+\sqrt{(2w^R-\boldsymbol{w})^2+\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}}\right).
\end{aligned}
$$

The first equation has no solution since s is nonnegative and the right-hand side is negative. Now we can write the thermal limit constraint as:

$$
(w^I)^2\leqslant\frac{\boldsymbol{w}}{2}\left(2w^R-\boldsymbol{w}+\sqrt{(2w^R-\boldsymbol{w})^2+\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}}\right)-(w^R)^2.
$$

Let $R=\sqrt{(2w^R-\boldsymbol{w})^2+\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}}$ and $\phi(w^R)=\frac{\boldsymbol{w}}{2}(2w^R-\boldsymbol{w}+R)-(w^R)^2$. Constraint (3.14c) describes a convex set if $\phi(w^R)$ is concave. To obtain the conditions for its concavity, we will calculate the second derivative:

$$
\begin{aligned}
\phi'(w^R)&=\frac{\boldsymbol{w}}{2}(2+R')-2w^R,\\
\phi''(w^R)&=\frac{\boldsymbol{w}}{2}R''-2=\frac{\boldsymbol{w}}{2}\frac{4R-\frac{4(2w^R-\boldsymbol{w})^2}{R}}{R^2}-2.
\end{aligned}
$$

A function is concave if its second derivative is negative:

$$
\begin{aligned}
&\frac{\boldsymbol{w}}{2}\frac{4R-\frac{4(2w^R-\boldsymbol{w})^2}{R}}{R^2}-2<0\ \Leftrightarrow\\
&R^2-(2w^R-\boldsymbol{w})^2<\frac{R^3}{\boldsymbol{w}}.
\end{aligned}
$$

Observe that, from the definition of R, the left hand side of this inequality is equal to $\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}$:

$$
\begin{aligned}
&\frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|} < \frac{R^3}{\boldsymbol{w}} \;\Leftrightarrow\; \left(\frac{4\boldsymbol{s}^u\boldsymbol{w}}{|\boldsymbol{Y}|}\right)^{\frac{2}{3}} < R^2 \;\Leftrightarrow \\
&\sqrt[3]{\left(\frac{4\boldsymbol{s}^u\boldsymbol{w}}{|\boldsymbol{Y}|}\right)^2} < (2w^R - \boldsymbol{w})^2 + \frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|} \;\Leftrightarrow \\
&w^R > \frac{1}{2}\sqrt{\sqrt[3]{\left(\frac{4\boldsymbol{s}^u\boldsymbol{w}}{|\boldsymbol{Y}|}\right)^2} - \frac{4\boldsymbol{s}^u}{|\boldsymbol{Y}|}} + \frac{\boldsymbol{w}}{2}.
\end{aligned}
$$

Let $\psi(x) = x^{\frac{2}{3}}\boldsymbol{w}^{\frac{2}{3}} - x$. Find the stationary point of $\psi(x)$:

$$
\begin{aligned}
&\psi'(x) = \boldsymbol{w}^{\frac{2}{3}}\frac{2}{3}\frac{1}{\sqrt[3]{x}} - 1 = 0 \;\Leftrightarrow \\
&x = \frac{8\boldsymbol{w}^2}{27}
\end{aligned}
$$

To verify the second order optimality condition, calculate the second derivative:

$$
\psi''(x) = -\frac{2}{9}\boldsymbol{w}^{\frac{2}{3}}\frac{1}{\sqrt[3]{x^4}} < 0.
$$

Hence $\psi(x)$ is concave, $x = \frac{8\boldsymbol{w}^2}{27}$ is a maximum and

$$
\psi(\frac{8\boldsymbol{w}^2}{27}) = (\frac{8\boldsymbol{w}^2}{27})^{\frac{2}{3}}\boldsymbol{w}^{\frac{2}{3}} - \frac{8\boldsymbol{w}^2}{27} = \frac{4\boldsymbol{w}^2}{27}.
$$

We have shown that $\psi(x) \leqslant \frac{4\boldsymbol{w}^2}{27}$ $\forall x > 0$. Then we can guarantee that constraint (3.14c) is convex if

$$
w^R > \frac{1}{2}\sqrt{\psi^{max}} + \frac{\boldsymbol{w}}{2} = \boldsymbol{w}(\frac{1}{3\sqrt{3}} + \frac{1}{2}).
$$

Since $(\frac{1}{3\sqrt{3}} + \frac{1}{2}) < 0.77$ and, by Lemma 3.14, all (w^R, w^I) such that $w^R < 0.77\boldsymbol{w}$ are infeasible, (3.14c) is convex everywhere where it is nonredundant.

□

Corollary 3.1. *Model 3.2 is boundary-invex.*

Proof. Based on Lemmas 3.15-3.18, we can show that all KKT points for the auxiliary problems (NLP_{2i}) are infeasible with respect to Model 3.2. Based on Definition 3.9, boundary-invexity is established.

□

3.9 Conclusion

Given a nonconvex optimisation problem, boundary-invexity captures the behaviour of the objective function on the boundary of its feasible region. In this work, we show that boundary-invexity is a necessary condition for KT-invexity, that becomes sufficient in the two-dimensional case. Unlike conventional invexity conditions, boundary-invexity can be verified algorithmically and, in some cases, in polynomial time.

Chapter 4

Semidefinite Programming Cuts

4.1 Introduction

In Chapter 3 we studied problems that are characterised by special properties that make them solvable to global optimality by polynomial time algorithms. In the remaining part of the thesis we consider more general nonconvex problems and focus on developing and solving convex relaxations, which can be used to determine how far away from the global optimum a local solution is or as part of global optimisation algorithms. This chapter deals with continuous nonconvex problems and concentrates on efficiently solving semidefinite programming relaxations.

SDP formulations are widely used as convex relaxations of nonconvex problems due to their ability to provide good quality bounds in many cases. The downside of this approach is the computational expensiveness of SDP, especially when the matrices required to be positive semidefinite become large. We propose an algorithm that uses tree decomposition to partition the underlying graph into sets of nodes (referred to as "tree decomposition bags") and then generates cuts iteratively. Given a solution of the previous iteration, for each tree decomposition bag it solves an SDP problem in order to find the best linear cut.

This chapter has the following structure. In Section 4.2 we provide a review of related literature. First, we recall some notions from graph theory that will be used in the subsequent sections. Then we give a review of methods used to solve SDP programs, including classical SDP algorithms, decomposition techniques and other SDP methods that have been applied to OPF. The last part of the section reviews linear cut generation algorithms.

Next follow the sections that describe our contributions. Section 4.3 considers the impact of linear cuts on the search space and derives the valid cut that maximises constraint violation at the solution of a relaxation that needs to be separated from the feasible set. In Section 4.4 we present our iterative cut generation algorithm and an improved polynomial relaxation of SDP problems. Section 4.5 shows how to apply this algorithm to the OPF problem. In Section 4.6, we present and discuss the results of the computational experiments. Section 4.7 concludes the chapter.

4.2 Background

4.2.1 Graph-theoretic background

In this section we provide a brief overview of graph-theoretic concepts that will be used further in this chapter.

Consider an undirected graph $G = (V, E)$ with the set of vertices V and the set of edges $E \subseteq V \times V$.

Two vertices v_i, v_j are called *adjacent* if they are connected by an edge $(v_i, v_j) \in E$. A graph is said to be *complete* if every two of its vertices are adjacent. Given a set of nodes $V' \subset V$, the *induced subgraph* on V' is the graph $G' = (V', E')$, where $E' = E \cap V' \times V'$. A *clique* is a subset of vertices whose induced subgraph is complete, and a *maximal clique* is a clique which cannot be extended by adding more vertices to it, i.e. it is not a subset of another clique.

A graph is *chordal* if every cycle of length $\geqslant 4$ has a chord (an edge connecting two vertices of a cycle that is itself not part of the cycle). A graph $G'(V, E')$ is a *chordal extension* of a graph $G(V, E)$ if it is chordal and $E \subset E'$.

Closely related to chordal graphs is the notion of *tree decomposition*:

Definition 4.1. *[165] A tree decomposition of a graph $G(V, E)$ is a family $(X_b : b \in B)$ of subsets of V, together with a tree $T(V^T, E^T)$ with $V^T = B$, with the following properties:*

1. $\bigcup_{b \in B} X_b = V$,
2. *Every edge $(i, j) \in E$ has both its ends in some X_b $(b \in B)$,*
3. *For $i, j, k \in B$, if j lies on the path of T from i to k then $X_i \cap X_k \subseteq X_j$.*

Each vertex in the tree T corresponds to a clique of the chordal extension of the graph G [74]. Tree decomposition of a given graph is not unique. The *width* of a tree decomposition is the cardinality of its largest clique, and the *treewidth* of a graph is the minimum width among all of its possible tree decompositions.

4.2.2 Semidefinite programming methods

Consider a convex nonlinear SDP problem in the form

$$\begin{aligned} \min \ & f(\mathbf{x}) \\ \text{s.t.} \ & g_i(\mathbf{x}) \leqslant 0, \ i = 1, \ldots, m, \\ & h_j(\mathbf{x}) = 0, \ j = 1, \ldots, k, \\ & X(\mathbf{x}) \succeq 0, \ \mathbf{x} \in \mathbb{R}^n, \end{aligned} \tag{SDP}$$

where f, g_i, $i = 1 \ldots, m$ and h_j, $j = 1, \ldots, k$ are twice differentiable functions and the feasible region is convex.

The interest in semidefinite programming has largely been motivated by its successful applications to combinatorial optimisation problems such as, for example, maximum cut and satisfiability [77], k-partitioning [66, 108], colouring [108] and betweenness problems [38] as well as for constructing convex relaxations of continuous nonconvex problems [67, 144, 119, 8]. Interior point methods have been the most commonly used technique for solving semidefinite programs after it was shown independently by Nesterov and Nemirovskii [146, 149, 147, 148, 145], Kamath and Karmarkar [106, 107] and Alizadeh [3] that their applications can be extended from linear to semidefinite programming. Since SDP problems are convex, these methods guarantee convergence to the global optimum. A detailed review of interior point methods for SDP can be found in a survey paper by Vandenberghe and Boyd [184].

Despite being solvable to global optimality by interior point methods, SDP problems still suffer from lack of scalability due to the fact that the state of the art SDP solvers become inefficient as matrix size increases.

This has prompted researchers to explore alternatives to interior point methods for semidefinite programming. Block coordinate descent methods [127] operate by partitioning the set of variables into blocks and minimising the objective function with respect to each block while the others are fixed. Block coordinate descent methods tailored for SDP [12] make use of the connection between positive semidefiniteness of a matrix and the properties of the Schur complement of one of its blocks. To ensure convergence in the presence of general constraints, these methods are usually applied in conjunction with augmented Lagrangian approaches [29, 28, 197]. In a similar spirit, the alternating direction method of multiplies [72, 76] optimises the augmented Lagrangian function by partitioning the problem. Adaptations of this algorithm to semidefinite programming were presented in works by Yu [194], Wen et al. [189] and Fukuda and Lourenço [68]. Madani et al. [129] applied the alternating direction methods of multipliers to SDP relaxations of OPF.

Other techniques that have been applied to SDP problems include regularisation methods [130, 159] and primal proximal methods [134]. Several specialised algorithms have been proposed for the problem of finding the projection of a point on the intersection of a cone and an affine surface [97].

A number of successful algorithms for SDP optimisation are based on nonsmooth optimisation techniques known as bundle methods. A large amount of theory in this area has been developed for SDP problems formulated as eigenvalue optimisation problems:

$$\min_{\mathbf{y}\in\mathbb{R}^m} \left(\lambda_{\max}(C - A^T\mathbf{y}) + \mathbf{b}^T\mathbf{y}\right),$$

where C is a symmetric $n \times n$ matrix, A is an $m \times n$ matrix, $\mathbf{b} \in \mathbb{R}^m$ and $\lambda_{\max}(X)$ denotes the maximum eigenvalue of matrix X. A number of SDP problems can be brought to the form for which the above model is the dual. Nonsmooth optimisation methods can also be applied to more general SDP problems.

Bundle algorithms use the information about the objective function and its subgradients from the current as well as previous iterations, referred to as a bundle, in order to construct piecewise linear approximations of the objective function and determine descent directions.

The algorithms that develop this general idea include proximal bundle methods [111, 169], spectral bundle methods [96] which use the special structure existing in eigenvalue problems, mixed polyhedral-semidefinite methods [191] and second order methods [170, 153, 95].

Surveys of the literature on various semidefinite programming methods can be found in the books by El Ghaoui and Niculescu [55] and Wolkowicz et al. [191].

Rewriting the SDP program as a nonlinear program by using the principal minors characterisation of PSD matrices and directly applying general nonlinear programming interior point methods was suggested by Hijazi et al. [99]. It uses the fact that a matrix is positive semidefinite if and only if all its principal minors are nonnegative. Although some of the resulting constraints are nonconvex, the feasible set is convex as long as for each matrix whose determinant is required to be nonnegative all its principal minors are required to be nonnegative as well. The advantage of this approach is that it allows solving SDP problems with efficient nonlinear programming solvers.

In order to reduce the computational effort, one can iteratively add violated SDP constraints to the model instead of including them all at once. The nonlinear principal minor constraints imposed upon the decomposed graph, as proposed by Hijazi et al. [99], can be treated as cuts to be added to the model dynamically. In the work by Miao et al. [138], the cuts were obtained by linearising the SDP constraints. Kocuk et al. [112, 113] used SDP cuts to strengthen their relaxations of OPF and OTS problems.

Other methods that have been proposed to improve the performance of SDP programming include exploiting sparsity, using alternative characterisations of the SDP conditions, generating linear outer approximations and various combinations of these.

Exploiting sparsity

Many SDP problems are described by partial matrices, i.e. matrices where some entries are specified and some are free. In such cases the notion of positive semidefiniteness is replaced by that of positive semidefinite completability: we cannot directly check the former because we do not have the full matrix, and the question becomes: can such a set of values be found that when assigned to the free entries, the matrix thus completed becomes PSD? If yes, the partial matrix is called PSD completable.

The sparsity-exploiting approaches are based on a general result about PSD completions of partial Hermitian matrices proved by Grone et al. [84]. Let $G = (V, E)$ be a finite undirected graph with the set of vertices V and the set of edges E and let $A(G)$ be a matrix where an element a_{ij} is defined if and only if $(i, j) \in E$. G is referred to as the sparsity pattern of the matrix $A(G)$.

Theorem 4.1. *[84] If all diagonal entries of $A(G)$ are specified, all principal minors consisting of specified entries are PSD and G is chordal, then $A(G)$ is PSD completable.*

This theorem requires the sparsity pattern graph to be chordal. In applications where this does not hold, tree decomposition (sometimes also referred to as Cholesky factorisation) is applied to obtain a chordal extension of the graph. A useful property of tree decomposition is that it finds all the cliques (subgraphs where every two vertices are connected by an edge) of a chordal completion of a given graph. These cliques are referred to as 'bags'. Once a set of bags is identified, it is enough to require that their corresponding matrices are PSD:

Theorem 4.2. *[69] If all diagonal entries of $A(G)$ are specified and all principal minors corresponding to the tree decomposition bags are PSD completable, then $A(G)$ is PSD completable.*

Using this result in the context of semidefinite programming was first proposed by Fukuda et al. [69]. Continuing this work, Nakata et al. [142] implemented the approach in two ways: by decomposing the matrix in the problem definition and by incorporating matrix completion into the interior point method itself. This approach is particularly promising for energy systems applications since the graphs representing the structure of power networks tend to have small treewidth. Tree decomposition has been applied to efficiently solve the SDP relaxation of OPF [7, 128].

SDP constraints as nonlinear cuts

As observed by Hijazi et al. [99], an SDP problem can be reformulated as a nonlinear programming problem with polynomial constraints. To do so, given an SDP condition of the form $X \succeq 0$, generalised Sylvester's criterion for positive semidefinite matrices is applied:

Theorem 4.3. *[26] A Hermitian matrix is positive semidefinite if and only if all its principal minors are nonnegative.*

Thus the constraint $X \succeq 0$ imposed on an $n \times n$ matrix is equivalent to the following set of polynomial inequalities:

$$\det(X_I) \geqslant 0 \quad \forall I \subseteq \{1, \ldots, n\},$$

where X_I denotes the submatrix of X obtained by removing all rows and columns except from those whose indices are contained in I.

$$\begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & \boldsymbol{x_{33}} \end{pmatrix} \qquad \begin{pmatrix} \boldsymbol{x_{11}} & \boldsymbol{x_{12}} & x_{13} \\ \boldsymbol{x_{21}} & \boldsymbol{x_{22}} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix} \qquad \begin{pmatrix} \boldsymbol{x_{11}} & x_{12} & \boldsymbol{x_{13}} \\ x_{21} & x_{22} & x_{23} \\ \boldsymbol{x_{31}} & x_{32} & \boldsymbol{x_{33}} \end{pmatrix}$$

I={3} I={1,2} I={1,3}

Figure 4.1: Examples of principal submatrices of a 3×3 matrix.

For example, for a matrix of size 3 with elements x_{ij}, $i, j = 1, 2, 3$, the polynomial constraints are written as:

$$\begin{aligned} & x_{11}(x_{22}x_{33} - x_{23}x_{32}) - x_{12}(x_{21}x_{33} - x_{23}x_{31}) \\ & \quad + x_{13}(x_{21}x_{32} - x_{22}x_{31}) \geqslant 0, \end{aligned} \tag{4.1a}$$

$$x_{11}x_{22} - x_{12}x_{21} \geqslant 0, \tag{4.1b}$$

$$x_{11}x_{33} - x_{13}x_{31} \geqslant 0, \tag{4.1c}$$

$$x_{22}x_{33} - x_{23}x_{32} \geqslant 0, \tag{4.1d}$$

$$x_{11} \geqslant 0,\ \ x_{22} \geqslant 0,\ \ x_{33} \geqslant 0. \tag{4.1e}$$

The polynomial reformulation allows SDP problems to be solved with the use of NLP solvers, which can often improve performance and numerical stability.

Although some of the polynomial constraints are nonconvex, the resulting set is convex as long as if a determinant constraint for some matrix is added to the model, all determinant constraints for its principal minors are included as well. This follows directly from the fact that the set of positive semidefinite matrices is convex. As Lasserre [120, 121] has shown, if a nondegeneracy assumption holds for a convex problem with a nonconvex algebraic representation, then every KKT point is a global optimum and methods that use a log-barrier function converge to the optimum.

This approach benefits from the decomposition methods discussed above. Since the number of polynomial constraints to be added is exponential in the size of the matrix, reducing matrix sizes leads to significant reductions of the resulting nonlinear program size.

Other approaches for semidefinite programming in OPF.

Recall that the SDP relaxation of OPF is obtained by disregarding the rank 1 constraint. If the solution has high rank, it can only provide a lower bound on the optimal objective function value but does not yield a physically feasible network configuration. To improve the quality and practical applicability of solutions, methods that search for low rank solutions can be applied.

Sojoudi et al. [174] developed a heuristic for converting low rank solutions of the SDP relaxation to rank-1 solutions. This method was shown to work on networks with cycles of size 3. Louca et al. [126] proposed minimising the approximated rank and solved this nonconvex problem using an iterative linearisation-minimisation algorithm.

Another vein of work focuses on using moment relaxations [119]. Unlike heuristics, they converge to the global optimum. Several works have applied this approach to OPF [140, 105, 75], and Molzahn and Hiskens [141] further developed it by exploiting sparsity. However, this method is computationally expensive and since the OPF problem is NP-hard, the order of the Lasserre hierarchy can become arbitrarily large.

Recently a coordinate descent algorithm that leads to low rank solutions was presented by Marecek and Martin [132].

4.2.3 Linear cut generation

The SDP constraints described in the previous section are given by computationally challenging matrix conditions or polynomial nonconvex inequalities. However, since the set is convex even though its algebraic representation is not, the constraints can be linearised. An infinite number of linear inequalities is necessary in order to accurately describe a convex nonlinear set, but in practice a finite number of linear cuts suffices when added by an iterative algorithm. The key is to generate only those cuts that will affect the optimal solution. Another strength of this approach is that it benefits from the fact that typically, not all nonlinear constraints are active at the optimal solution of a constrained optimisation problem.

Consider a general optimisation problem (P) that contains possibly nonconvex nonlinear constraints, which may include SDP constraints. Some or all variables in (P) may be discrete. Let $F(P)$ denote the feasible set of (P) and assume that $F(P)$ is convex.

The idea of constructing linear cuts to approximate feasible regions first appears in the cutting plane method introduced by Gomory for solving integer programs [80]. In the context of continuous convex programming, linear constraint generation was first implemented in the cutting plane method by Kelley [110] and Cheney and Goldstein [37] and was further develped by Veinott [185] and Wolfe [190]. The algorithm is based on replacing convex hypersurfaces with their linear approximations which are added iteratively. A similar approach is applied in the case where (P) is a mixed-integer nonlinear program in the Outer Approximation algorithm which approximates the continuous relaxation of (P) with linear constraints. Sequential Linear Programming methods [166, 60, 30, 164] (also known as successive linear programming or approximation programming methods) use first order Taylor approximations to linearise the models. Unlike the cutting plane approaches listed above, Sequential LP algorithms employ approximations instead of relaxations of the feasible set, and global convergence is ensured by the use of trust region strategies [46].

Cuts can be divided into optimality and feasibility cuts. Optimality cuts exclude from the search space those subregions of the feasible set where the objective function value is provably larger (for a minimisation problem) than at a known feasible solution and are used, for example, in Benders decomposition [20, 73]. Feasibility cuts function as linear relaxations of the nonlinear constraints and remove only those points which are known to be infeasible, that is, they are valid with respect to the nonlinear feasible set $F(P)$. We will focus on feasibility cuts.

The choice of points at which the approximations are built is important for the validity and performance of the algorithms. Many methods use the solution of a relaxation as the point around which the approximations are built. For example, given a convex nonlinear constraint $g(\mathbf{x}) \leqslant 0$ and a point $\mathbf{x}^*$ which violates this constraint, one can build the following separating hyperplane:

$$g(\mathbf{x}^*) + \nabla g(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \leqslant 0. \tag{4.2}$$

The above is guaranteed to be a valid cut only if the function g is convex. If the algebraic representation is nonconvex, then the linear cut might remove feasible points $\mathbf{x} \in F(P)$ even if the feasible set $F(P)$ is convex (an example is given by Fig. 4.2). Bonami et al. [23] have modified this method by choosing $\mathbf{x}^*$ to be on the boundary of the feasible set. They prove that this leads to valid cuts:

Lemma 4.1. *[23] If the feasible set $F(P)$ is convex and $g(\mathbf{x}^*) = 0$, then 4.2 is valid for*

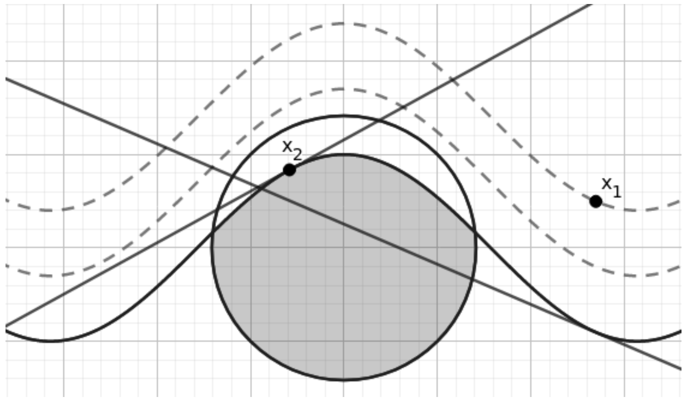


Figure 4.2: Linear cuts (4.2) for a convex feasible region (shown in grey). The cut with a strictly infeasible parameter $\mathbf{x}^* = \mathbf{x}_1$ is invalid, while the cut through a boundary point $\mathbf{x}^* = \mathbf{x}_2$ is valid.

$F(P)$.

Suppose that $F(P)$ is convex and an infeasible point $\mathbf{x}^* \notin F(P)$ was found by solving a relaxation. To obtain a point on the boundary, the algorithm proposed by Bonami et al. [23] solves a subproblem that finds the nearest (in terms of the l_2-norm) point with respect to $\mathbf{x}^*$:

$$\begin{aligned} &\min ||\mathbf{x} - \mathbf{x}^*|| \\ &s.t.\ \mathbf{x} \in F_0, \end{aligned} \tag{ProjNLP}$$

where F_0 denotes the set to be approximated. In practice this problem is constructed to be more tractable than the original problem (P). For example, in the outer approximation method for MINLPs [23] F_0 represents the feasible set of the continuous relaxation, and in continuous applications it can be defined by a subset of constraints of the original problem (P), as we will discuss further in this section in the context of SDP problems.

Given a solution $\hat{\mathbf{x}}$ of (ProjNLP), the cuts are generated either using the gradients of the active constraints according to (4.2) or by generating a hyperplane that is orthogonal to the vector $(\mathbf{x}^* - \hat{\mathbf{x}})$ [23]:

$$(\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0. \tag{4.3}$$

The validity of the above cut is shown, for example, in the proof of the separating hyperplane theorem [21].

The objective in (ProjNLP) does not necessarily have to be written in terms of the l_2-norm. Hamzeei and Luedtke [90] proposed an algorithm that allows different choices of the norm. They also compared the performance of MINLP algorithms based on gradient cuts (4.2) and orthogonal cuts (4.3) and made a conclusion that the gradient cuts lead to better performance. Orthogonal cuts, however, do not require computing the gradients, which is beneficial for those applications where these computations are expensive.

In addition to being valid for convex sets with nonconvex representations, cuts generated at the boundary often result in larger reductions of the search space. Indeed, if a cut does not pass through any point on the boundary of the feasible set, it can be made tighter by projecting it onto this surface, albeit at the cost of solving the projection subproblem.

Linear cuts for SDP

Linear cut generation approaches have been successfully applied to semidefinite programs. The polyhedral cutting plane method [116, 79, 114] uses the semi-infinite programming reformulation of (SDP) where the constraint $X(\mathbf{x}) \succeq 0$ is rewritten as

$$\mathbf{d}^T X(\mathbf{x})\mathbf{d} \geqslant 0 \ \forall \mathbf{d} \text{ such that } ||\mathbf{d}|| = 1.$$

The linear cuts are constructed by relaxing the above condition. First, a solution $\mathbf{x}^*$ of a relaxation of (SDP) that ignores the SDP constraint $X(\mathbf{x}) \succeq 0$ is obtained. A vector $\mathbf{d}_{neg}$ such that $\mathbf{d}_{neg}^T X(\mathbf{x})\mathbf{d}_{neg} < 0$ is found and the following linear cut is added to the problem:

$$\mathbf{d}_{neg}^T X(\mathbf{x})\mathbf{d}_{neg} \geqslant 0.$$

$\mathbf{d}_{neg}$ may be chosen as the eigenvector corresponding to the smallest eigenvalue of $\mathbf{X}(\mathbf{x}^*)$, which can be found by solving the subproblem

$$\begin{aligned} &\min_{\mathbf{d}} \mathbf{d}^T X(\mathbf{x}^*)\mathbf{d} \\ &s.t. \ ||\mathbf{d}|| = 1. \end{aligned}$$

A recent work on solving the SDP relaxation of OPF [138] generates cuts in the form (4.3).

4.3 Deepest Valid Cut

In this Section we define the notion of the deepest cut and prove that linear cuts in the form (4.3) satisfy this definition.

Consider a convex set F_0 which may have a nonconvex algebraic representation and a point $\mathbf{x}^* \notin F_0$. There exists an infinite number of linear cuts that would separate $\mathbf{x}^*$ from F_0, and the question is which cut will yield the largest improvement of the relaxation quality. One possible measure of this is the signed Euclidean distance from $\mathbf{x}^*$ to the surface of the cut expressed as $\frac{\mathbf{a}^T\mathbf{x}^* - b}{||\mathbf{a}||}$, which reflects scaled constraint violation [52]. In this section we prove that the orthogonal cut (4.3) maximises this number.

Definition 4.2. *(Deepest valid cut) Given a convex set F and a point $\mathbf{x}^* \notin F$, the deepest valid cut is the linear cut $\mathbf{a}^T\mathbf{x} - b \leqslant 0$ that maximises the scaled violation at $\mathbf{x}^*$ which is calculated as $\frac{\mathbf{a}^T\mathbf{x}^* - b}{||\mathbf{a}||}$ and is a valid constraint for F.*

We start by identifying a condition that is necessary for the cut to be valid.

Lemma 4.2. *Consider the solution of (ProjNLP) $\hat{\mathbf{x}}$ and a linear cut of the form $\mathbf{a}^T\mathbf{x} - b \leqslant 0$. If this cut is a valid cut for F, then $b \geqslant \mathbf{a}^T\hat{\mathbf{x}}$.*

Proof. Since $\hat{\mathbf{x}} \in F$, $\hat{\mathbf{x}}$ is feasible with respect to any valid cut:

$$\mathbf{a}^T\hat{\mathbf{x}} - b \leqslant 0 \;\Rightarrow\; b \geqslant \mathbf{a}^T\hat{\mathbf{x}}.$$

□

We will now find the deepest cut that satisfies this condition.

Lemma 4.3. *Given a set of linear cuts of the form $\mathbf{a}^T\mathbf{x} - b \leqslant 0$ with $b \geqslant \mathbf{a}^T\hat{\mathbf{x}}$, the deepest cut with respect to $\mathbf{x}^*$ is given by $(\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0$.*

Proof. Consider a problem of maximizing the scaled signed Euclidean distance from $\mathbf{x}^*$ to the cut surface over all cuts of the form $\mathbf{a}^T\mathbf{x} - b \leqslant 0$ with $b \geqslant \mathbf{a}^T\hat{\mathbf{x}}$:

$$\max_{\mathbf{a},b} \; f(\mathbf{a}, b) = \frac{\mathbf{a}^T\mathbf{x}^* - b}{||\mathbf{a}||} \text{ s.t. } b \geqslant \mathbf{a}^T\hat{\mathbf{x}}.$$

Given any fixed $\mathbf{a}$, $b = \mathbf{a}^T\hat{\mathbf{x}}$ maximises f. This allows us to rewrite the problem as:

$$\max \; f(\mathbf{a}) = \frac{\mathbf{a}^T(\mathbf{x}^* - \hat{\mathbf{x}})}{||\mathbf{a}||}.$$

Recalling that a scalar product can be expressed as a product of the norms of the vectors and the cosine of the angle between them, the above is equivalent to:

$$\max_{\mathbf{a},\phi} \frac{||\mathbf{a}|| \cdot ||\mathbf{x}^* - \hat{\mathbf{x}}|| \cos\phi}{||\mathbf{a}||} = ||\mathbf{x}^* - \hat{\mathbf{x}}|| \cos\phi,$$

where ϕ is the angle between $\mathbf{a}$ and $(\mathbf{x}^* - \hat{\mathbf{x}})$. Since $||\mathbf{x}^* - \hat{\mathbf{x}}||$ is constant, this expression depends on ϕ only and its maximum is attained at $\phi = 0$. The angle between optimal $\mathbf{a}$ and $\overline{\hat{\mathbf{x}}\mathbf{x}^*}$ being zero implies that $\mathbf{a}$ is a positive scalar multiple of $(\mathbf{x}^* - \hat{\mathbf{x}})$, i.e. the optimal value of $\mathbf{a}$ is $c(\mathbf{x}^* - \hat{\mathbf{x}})$, where c is an arbitrary positive constant. Thus the best cut is given by:

$$c(\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0 \;\Leftrightarrow\; (\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0.$$

□

The problem solved in the above lemma is a relaxation of the problem of finding the deepest valid cut. However, it can be observed that the deepest valid cut obtained is exactly the orthogonal cut (4.3). This implies the following result:

Theorem 4.4. *Given a convex set F_0 and a point $\mathbf{x}^* \notin F_0$, the deepest valid cut is given by $(\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0$, where $\hat{\mathbf{x}}$ is the solution of the projection problem (ProjNLP).*

Proof. Follows from Lemma 4.3 and the validity of cut (4.3). □

An example of a linear cut built using Theorem 4.4 is shown in Figure 4.3.

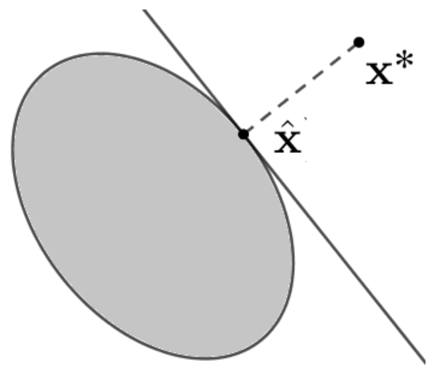


Figure 4.3: Deepest valid cut. The nonlinear feasible set is shown in grey colour, the feasible set of the linear cut is positioned to the left and below the solid line.

4.4 Improved Cuts for SDP Problems

In this section we propose two improved cut generation approaches for SDP problems. Building upon Theorem 4.4 which provides the procedure for finding the deepest valid cut, we present an algorithm that combines graph partitioning, linearisation and dynamic constraint generation techniques in an iterative process. After that we develop a strengthened variation of the polynomial cuts that were introduced by Hijazi et al. [99] and described in Subsection 4.2.2 of this chapter.

Consider a problem of the form:

$$\begin{aligned}
&\min f(\mathbf{x}, \mathbf{y}) \\
&\text{s.t.} \\
&g_i(\mathbf{x}, \mathbf{y}) \leqslant 0 \ \forall i \in 1, \ldots, p, \\
&h_j(\mathbf{x}, \mathbf{y}) = 0 \ \forall j \in 1, \ldots, q, \\
&X(\mathbf{x}) \succeq 0, \\
&\mathbf{x}^l \leqslant \mathbf{x} \leqslant \mathbf{x}^u, \ \mathbf{y}^l \leqslant \mathbf{y} \leqslant \mathbf{y}^u, \\
&\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m,
\end{aligned} \tag{SDP}$$

where functions f, g_i, $i \in 1, \ldots, p$ and h_j, $j \in 1, \ldots, q$ are twice continuously differentiable and X is a matrix whose entries are expressions of $\mathbf{x}$.

4.4.1 Tree decomposition heuristic

Let the graph $G = (V, E)$ be the sparsity pattern of matrix $X(\mathbf{x})$. $B(X(\mathbf{x}))$ will denote the set of bags of a tree decomposition of G, where each element $b \in B(X(\mathbf{x}))$ is a subset of vertices of G: $b \subseteq V$.

Theorem 4.2 implies that constraint $X(\mathbf{x}) \succeq 0$ can be replaced by a set of constraints that correspond to each bag $b \in B(X(\mathbf{x}))$. Thus model (SDP) is equivalent to:

$$
\begin{aligned}
&\min f(\mathbf{x}, \mathbf{y}) \\
&\text{s.t.} \\
&g_i(\mathbf{x}, \mathbf{y}) \leqslant 0 \ \forall i \in 1, \ldots, p, \\
&h_j(\mathbf{x}, \mathbf{y}) = 0 \ \forall j \in 1, \ldots, q, \\
&X_b(\mathbf{x}) \succeq 0 \ \forall b \in B(X(\mathbf{x})), \\
&\mathbf{x}^l \leqslant \mathbf{x} \leqslant \mathbf{x}^u, \ \mathbf{y}^l \leqslant \mathbf{y} \leqslant \mathbf{y}^u, \\
&\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m.
\end{aligned}
\tag{SDP\_D}
$$

We are interested in obtaining the tree decomposition with minimal width. This problem is NP-hard, but efficient heuristic algorithms that approximate such a decomposition exist. We are using the greedy fill-in algorithm [89] to perform the tree decomposition of graph G and obtain the set of bags $B(X(\mathbf{x}))$.

First we shall define the notion of fill-in. Consider a vertex v and a subgraph G^v induced by v and its adjacent vertices. The fill-in of v is the number of edges that need to be added to G^v in order for it to become a clique. Now we can write the tree decomposition algorithm:

Algorithm 1 The greedy fill-in tree decomposition algorithm

```
Let G'(V', E') = G(V, E)
Initialise T as an empty graph
while |V'| > 1 do
    Choose a vertex u with minimal fill-in
    Let G'^u be the subgraph induced by u and its adjacent nodes
    Add the bag comprised of the vertices of G'^u as a vertex to T
    Update G' by connecting all vertices in G'^u and removing u
end while
```

4.4.2 Dynamic linear cut generation

Here we introduce a dynamic linear cut generation approach, adding cuts corresponding to violated SDP constraints in an iterative process.

Generating the linear cut Consider a tree decomposition bag $b \in B(X(\mathbf{x}))$. Let F_b denote the set of all vectors $\mathbf{x}$ that satisfy the SDP conditions for b and the bound constraints:

$$F_b = \left\{ \mathbf{x} \in \mathbb{R}^n \;\middle|\; \begin{array}{l} X_b(\mathbf{x}) \succeq 0, \\ \mathbf{x}^l \leqslant \mathbf{x} \leqslant \mathbf{x}^u. \end{array} \right\}$$

F_b is a convex set, which allows us to construct its outer approximations. Our method avoids explicitly computing the gradients of the polynomial principal minor constraints and instead generates an orthogonal cut that is guaranteed to be valid.

Consider a point $\mathbf{x}^*$ violating the SDP constraints for bag b: $\mathbf{x}^* \notin F_b$. We want to find the linear cut that would separate $\mathbf{x}^*$ from F_b, and cuts that are most violated by $\mathbf{x}^*$ are preferable. As in Definition 4.2, we will measure the constraint violation by the Euclidean distance from $\mathbf{x}^*$ to the surface of the cut.

By Theorem 4.4, in order to find the deepest cut, the point in F_b with the minimal distance from $\mathbf{x}^*$ needs to be found by solving the projection problem (ProjNLP) for bag b:

$$\min ||\mathbf{x}^* - \mathbf{x}|| \text{ s.t. } \mathbf{x} \in F_b. \tag{ProjNLP$_b$}$$

Let $\hat{\mathbf{x}}_b$ denote the solution of (ProjNLP$_b$). The linear cut that maximises scaled constraint violation at $\mathbf{x}^*$ is generated in the form (4.3):

$$(\mathbf{x}^* - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) \leqslant 0. \tag{4.4}$$

The cut generating algorithm Now we can describe the cut generation algorithm. First, the relaxed problem that does not include any SDP constraints is solved using a nonlinear programming solver:

$$\begin{aligned} &\min f(\mathbf{x}, \mathbf{y}) \\ &\text{s.t.} \\ &g_i(\mathbf{x}, \mathbf{y}) \leqslant 0 \ \forall i \in 1, \ldots, p, \\ &h_j(\mathbf{x}, \mathbf{y}) = 0 \ \forall j \in 1, \ldots, q, \\ &\mathbf{x}^l \leqslant \mathbf{x} \leqslant \mathbf{x}^u, \ \mathbf{y}^l \leqslant \mathbf{y} \leqslant \mathbf{y}^u. \end{aligned} \tag{SDP\_R}$$

After that, given a solution $\mathbf{x}^*_{k-1}$ of the previous iteration, SDP conditions are checked for each bag. If SDP conditions are violated for a bag b, an SDP solver is called to solve the (ProjNLP$_b$) problem for b and $\mathbf{x}^*_{k-1}$. Using its solution $\hat{\mathbf{x}}^k_b$, a linearisation of the SDP constraints is added for the corresponding bag according to formula (4.4). After updating the problem with all linear cuts generated by this method, the model is solved again. This process is repeated until a given termination condition is satisfied. The pseudocode for this method is given by Algorithm 2.

Algorithm 2 Dynamic linear cut generation

Solve (SDP_R)
while Termination condition not satisfied **do**
    for Each bag $b \in B(X(\mathbf{x}))$ **do**
        if $P_b(\mathbf{x}^*_{k-1}) \notin F_b$ **then**
            Solve (ProjNLP_b) for $P_b(\mathbf{x}^*_{k-1})$ and F_b to find $\hat{\mathbf{x}}^k_b \in \mathbb{R}^{n(b)}$
            Add cut $\left(P_b(\mathbf{x}^*_{k-1}) - \hat{\mathbf{x}}^k_b\right)^T \left(P_b(\mathbf{x}) - \hat{\mathbf{x}}^k_b\right) \leqslant 0$ to the model
        end if
    end for
    Solve the updated model
end while

Here $P_b(\mathbf{x}) \in \mathbb{R}^{n(b)}$ denotes the projection of $\mathbf{x}$ onto the variable space of bag b and $n(b)$ is the number of elements of $\mathbf{x}$ that appear in the entries of X_b. This notation is used to show that we can define each projection subproblem in terms of a small subset of problem variables.

4.4.3 Strengthened polynomial formulation

Hijazi et al. [99] applied the polynomial formulation (4.1) to tree decomposition bags of size 3 to obtain a relaxation of an SDP problem. We propose to strengthen this relaxation by adding polynomial constraints that require all principal minors of size 3 of larger bags to be nonnegative.

Consider problem (SDP_D) and, in particular, constraint

$$X_b(\mathbf{x}) \succeq 0 \ \forall b \in B(X(\mathbf{x})), \tag{4.5}$$

where $B(X(\mathbf{x}))$ is the set of all tree decomposition bags of $X(\mathbf{x})$. Let b_i denote a bag in $B(X(\mathbf{x}))$ such that $|b_i| = k > 3$ and consider the corresponding submatrix:

$$X_{b_i}(\mathbf{x}) = \begin{pmatrix} x_{i_1 i_1} & x_{i_1 i_2} & \dots & x_{i_1 i_k} \\ x_{i_2 i_1} & x_{i_2 i_2} & \dots & x_{i_2 i_k} \\ \dots & & & \\ x_{i_k i_1} & x_{i_k i_2} & \dots & x_{i_k i_k} \end{pmatrix},$$

where $\{i_1, \dots, i_k\} \subseteq \{1, \dots, n\}$. $X_{b_i}(\mathbf{x})$ has C^3_k (binomial coefficient) 3-dimensional principal submatrices $X_J(\mathbf{x})$, where $J \subset b_i$. Then by Theorem 4.3 we have

$$X_{b_i}(\mathbf{x}) \succeq 0 \ \Rightarrow \ X_J(\mathbf{x}) \succeq 0 \ \forall J \subset b_i, |J| = 3,$$

and a strengthened relaxation of constraint (4.5) involving only matrices of size less than 3×3 can be written as

$$X_{b_i}(\mathbf{x}) \succeq 0, \ \forall b_i \in B(X(\mathbf{x})) \text{ such that } |b_i| \leqslant 3,$$

$$X_J(\mathbf{x}) \succeq 0,\ \forall J \subset b_i,\ \forall b_i \in B(X(\mathbf{x})) \text{ such that } |b_i| > 3, |J| = 3.$$

Applying Theorem 4.3 allows to write the above constraints as polynomial inequalities:

$$\begin{aligned}
&\quad \text{For all } b_i \in B(X(\mathbf{x})) \text{ such that } |b_i| \leqslant 3:\\
&\det(X_{b_i}(\mathbf{x})) \geqslant 0,\\
&\det(X_{\{i,j\}}(\mathbf{x})) \geqslant 0,\ \forall i,j \subset b_i,\\
&X_{\{i\}}(\mathbf{x}) \geqslant 0,\ \forall i \in b_i;\\
&\quad \text{For all } J \subset b_i,\ \forall b_i \in B(X(\mathbf{x})) \text{ such that } |b_i| > 3,\ |J| = 3:\\
&\det(X_J(\mathbf{x})) \geqslant 0,\\
&\det(X_{\{i,j\}}(\mathbf{x})) \geqslant 0,\ \forall i,j \subset J,\\
&X_{\{i\}}(\mathbf{x}) \geqslant 0,\ \forall i \in J.
\end{aligned}$$

4.5 Application to Optimal Power Flow

In this Section we apply our findings to generate linearised SDP cuts for the Optimal Power Flow problem. Let us recall the formulation of the SDP relaxation of OPF that was given in Model 2.3:

Model 4.1: The Semidefinite Programming relaxation of the OPF problem

variables for each $(i,j) \in E$:

$p_{ij},\ q_{ij}$

variables for each $(i,j) \in N \times N$:

$w^R_{ij} \in [(\boldsymbol{w}^R_{ij})^l,\ (\boldsymbol{w}^R_{ij})^u],\ w^I_{ij} \in [(\boldsymbol{w}^I_{ij})^l,\ (\boldsymbol{w}^I_{ij})^u]$

variables for each $i \in N$:

$w_i \in [(\boldsymbol{v}^l_i)^2,\ (\boldsymbol{v}^u_i)^2]$

$p^g_i \in [\boldsymbol{p}^{gl}_i,\ \boldsymbol{p}^{gu}_i],\ q^g_i \in [\boldsymbol{q}^{gl}_i,\ \boldsymbol{q}^{gu}_i]$

objective:

$$\min \sum_{i \in N} \left(\boldsymbol{c}_{2i}(p^g_i)^2 + \boldsymbol{c}_{1i} p^g_i + \boldsymbol{c}_{0i}\right)$$

subject to:

$$p^g_i - \boldsymbol{p}^d_i = \sum_{(i,j)\in E} p_{ij} + \sum_{(j,i)\in E} p_{ij}\quad \forall i \in N \tag{4.6a}$$

$$q^g_i - \boldsymbol{q}^d_i = \sum_{(i,j)\in E} q_{ij} + \sum_{(j,i)\in E} q_{ij}\quad \forall i \in N \tag{4.6b}$$

$$p_{ij} = \mathbf{g}_{ij} w_i - \mathbf{g}_{ij} w^R_{ij} - \mathbf{b}_{ij} w^I_{ij}\quad \forall (i,j) \in E \tag{4.6c}$$

$$q_{ij} = -\mathbf{b}_{ij} w_i + \mathbf{b}_{ij} w^R_{ij} - \mathbf{g}_{ij} w^I_{ij}\quad \forall (i,j) \in E \tag{4.6d}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{s}^{\boldsymbol{u}}_{ij}\quad \forall (i,j) \in E \tag{4.6e}$$

$$w_{ij}^R \tan(\boldsymbol{\theta}_{ij}^l) \leqslant w_{ij}^I \leqslant w_{ij}^R \tan(\boldsymbol{\theta}_{ij}^u) \quad \forall (i,j) \in E \tag{4.6f}$$

$$W \succeq 0 \tag{4.6g}$$

---

where matrix W is defined as

$$W = \begin{cases} W_{ii} = w_i \ \forall i \in N, \\ W_{ij} = w_{ij}^R + \boldsymbol{i} w_{ij}^I \ \forall \ (i,j) \in E. \end{cases}$$

The sparsity pattern graph of matrix W is represented by the network graph $G = (N, E)$ consisting of a set of nodes N and a set of lines E. After applying tree decomposition to obtain the set of bags $B(W)$, constraint (4.6g) becomes:

$$W_b \succeq 0 \ \forall b \in B(W). \tag{4.7}$$

Madani et al. [128] showed that the treewidth (the size of the largest bag) is small for standard OPF benchmarks. The same should hold for most real life networks since they tend to be sparse. Smaller bags lead to smaller matrices which can be handled efficiently by the solvers.

Our algorithm starts by solving the SOCP relaxation of Model 4.1 obtained by replacing condition (4.6g) by the second-order cone constraint $(w_{ij}^R)^2 + (w_{ij}^I)^2 \leqslant w_i w_j$. Denote the solution of this problem by $\mathbf{x}_0^*$ and the solutions of problems obtained by adding linear cuts by $\mathbf{x}_k^*$, where k is the number of the iteration of the algorithm.

4.5.1 Bag matrix completion

The graphs induced by the tree decomposition bags $b \in B(W)$ are cliques in the chordal extension of the graph G but not in the original graph. Since the SOCP relaxation includes only the variables associated with the lines that exist in the network, the matrices corresponding to the bags are often partially defined. Let us first consider bags of size 3.

Suppose that a bag b contains nodes $\{1, 2, 3\}$. Its matrix has the form:

$$W_b = \begin{pmatrix} w_1 & w_{12}^R - \boldsymbol{i} w_{12}^I & w_{13}^R - \boldsymbol{i} w_{13}^I \\ w_{12}^R + \boldsymbol{i} w_{12}^I & w_2 & w_{23}^R - \boldsymbol{i} w_{23}^I \\ w_{13}^R + \boldsymbol{i} w_{13}^I & w_{23}^R + \boldsymbol{i} w_{23}^I & w_3 \end{pmatrix},$$

where the values of some off-diagonal elements might be unknown at a given solution $\mathbf{x}_k^*$.

Lemma 4.4. *For any bag b of size 3 the matrix W_b always has a positive semidefinite completion.*

Proof. Since only a graph containing cycles of length greater than 4 can be nonchordal, all graphs corresponding to bags of size 3 are chordal. By Theorem 4.1 this implies that the matrix W_b has a positive semidefinite completion if all minors corresponding to the maximal cliques in b are nonnegative. Since the squared voltages are always positive, any minor corresponding to a single node is positive. Now consider minors of size 2 which are written as:

$$w_i w_j - (w_{ij}^R - \boldsymbol{i} w_{ij}^I)(w_{ij}^R + \boldsymbol{i} w_{ij}^I) = w_i w_j - (w_{ij}^R)^2 - (w_{ij}^I)^2.$$

A solution $\mathbf{x}_k^*$ satisfies the second-order cone constraints, therefore the above is nonnegative for any clique of size 2. We need not check the minor of size 3 since it is equivalent to the determinant of W_b which is known only if W_b is fully defined.

Thus W_b is PSD-completable. □

The second-order cone constraints tend to be active at $\mathbf{x}_k^*$, that is,

$$(w_{ij}^R)^2 + (w_{ij}^I)^2 = w_i w_j.$$

Empiric observations show that this is the case for over 90% of lines in all PGLib instances. This gives us additional information about the matrices W_b.

Theorem 4.5. *Suppose that the values w_{ij}^R, w_{ij}^I associated with all but one pair of nodes in a set of nodes b_i of size 3 are known and at least one corresponding second-order cone constraint is tight. Then W_{b_i} has a unique positive semidefinite completion.*

Proof. Let the nodes in b_i be denoted as $\{1, 2, 3\}$. We will assume without loss of generality that the undefined line is line $(1, 2)$. For clarity purposes, let $x = w_{12}^R$ and $y = w_{12}^I$. If x and y yield a PSD completion of W_{b_i}, then the determinant of W_{b_i} is nonnegative:

$$\begin{aligned}2x(w_{13}^R w_{32}^R - w_{13}^I w_{32}^I) + 2y(w_{13}^R w_{32}^I + w_{32}^R w_{13}^I) - (x^2 + y^2) \geqslant \\ ((w_{13}^R)^2 + (w_{13}^I)^2)w_2 + ((w_{32}^R)^2 + (w_{32}^I)^2)w_1 - w_1 w_2 w_3.\end{aligned}$$

All variables except for x and y are treated here as constants since their values are known. Therefore the above inequality describes a circle in the (x, y) space. We will rewrite it in an equivalent form in order to find the centre and radius of the circle.

$$\begin{aligned}(-w_3 x^2 + 2(w_{13}^R w_{32}^R - w_{13}^I w_{32}^I)x) + (-w_3 y^2 + 2(w_{13}^R w_{32}^I + w_{32}^R w_{13}^I)y) \geqslant \\ ((w_{13}^R)^2 + (w_{13}^I)^2)w_2 + ((w_{32}^R)^2 + (w_{32}^I)^2)w_1 - w_1 w_2 w_3.\end{aligned}$$

By dividing the above by $-w_3$ we obtain:

$$\begin{aligned}(x^2 - \frac{2}{w_3}(w_{13}^R w_{32}^R - w_{13}^I w_{32}^I)x) + (y^2 - \frac{2}{w_3}(w_{13}^R w_{32}^I + w_{32}^R w_{13}^I)y) \leqslant \\ -((w_{13}^R)^2 + (w_{13}^I)^2)\frac{w_2}{w_3} - ((w_{32}^R)^2 + (w_{32}^I)^2)\frac{w_1}{w_3} + w_1 w_2.\end{aligned}$$

This is equivalent to:

$$(x - \frac{w_{13}^R w_{32}^R - w_{13}^I w_{32}^I}{w_3})^2 + (y - \frac{w_{13}^R w_{32}^I + w_{32}^R w_{13}^I}{w_3})^2 \leqslant$$

$$\frac{1}{w_3^2}((w_{13}^R w_{32}^R - w_{13}^I w_{32}^I)^2 + (w_{13}^R w_{32}^I + w_{32}^R w_{13}^I)^2) -$$
$$((w_{13}^R)^2 + (w_{13}^I)^2)\frac{w_2}{w_3} - ((w_{32}^R)^2 + (w_{32}^I)^2)\frac{w_1}{w_3} + w_1 w_2.$$

This describes a circle with the centre at:

$$\boldsymbol{x}^c = \frac{w_{13}^R w_{32}^R - w_{13}^I w_{32}^I}{w_3},$$
$$\boldsymbol{y}^c = \frac{w_{13}^R w_{32}^I + w_{32}^R w_{13}^I}{w_3}$$

and radius $\boldsymbol{R}$ such that

$$\boldsymbol{R}^2 = \frac{1}{w_3^2}\left((w_{13}^R w_{32}^R - w_{13}^I w_{32}^I)^2 + (w_{13}^R w_{32}^I + w_{32}^R w_{13}^I)^2\right) -$$
$$((w_{13}^R)^2 + (w_{13}^I)^2)\frac{w_2}{w_3} - ((w_{32}^R)^2 + (w_{32}^I)^2)\frac{w_1}{w_3} + w_1 w_2.$$

By expanding the squared sums we can show that the first term in $\boldsymbol{R}^2$ is equal to:

$$\frac{1}{w_3^2}\left((w_{32}^R w_{13}^R)^2 + (w_{32}^I w_{13}^I)^2 + (w_{32}^I w_{13}^R)^2 + (w_{32}^R w_{13}^I)^2\right)$$
$$= \frac{1}{w_3^2}\left((w_{13}^R)^2((w_{32}^R)^2 + (w_{32}^I)^2) + (w_{13}^I)^2((w_{32}^R)^2 + (w_{32}^I)^2)\right)$$

Let $S_{ij} = (w_{ij}^R)^2 + (w_{ij}^I)^2$. Then the above can be written as $\frac{S_{32}S_{13}}{w_3^2}$ and the radius is given by

$$\boldsymbol{R}^2 = \frac{S_{32}S_{13}}{w_3^2} - \frac{S_{13}w_2 + S_{32}w_1}{w_3} + w_1 w_2 =$$
$$\frac{S_{32}}{w_3}\frac{S_{13}}{w_3} - \frac{S_{13}w_2}{w_3} - \frac{S_{32}w_1}{w_3} + w_1 w_2 = \frac{S_{32}}{w_3}(\frac{S_{13}}{w_3} - w_1) - w_2(\frac{S_{13}}{w_3} - w_1) =$$
$$(\frac{S_{13}}{w_3} - w_1)(\frac{S_{32}}{w_3} - w_2) = \frac{1}{w_3^2}(S_{13} - w_1 w_3)(S_{32} - w_2 w_3).$$

Thus if at least one of the SOC constraints for lines $(1,3)$ and $(3,2)$ is active at a given solution, then the radius of the feasible circle is 0 and the SDP inequality has a unique solution (the point $(\boldsymbol{x}^c, \boldsymbol{y}^c)$). By Lemma 4.4, W_{b_i} is PSD completable, therefore $(\boldsymbol{x}^c, \boldsymbol{y}^c)$ is a valid PSD completion.

The proof is similar for the cases when line 13 or 32 is missing. □

In our implementation we assume that the conditions of Theorem 4.5 are satisfied for each three dimensional matrix with one unknown element. Although some bags might violate this condition, in the worst case the assumption will result in the addition of a small number of redundant linear constraints that will not affect the validity and performance of the algorithm.

Theorem 4.5 can be applied not only to tree decomposition bags of size 3 but also to subsets of nodes within larger bags ($|b| > 3$). Indeed, since all principal minors of a PSD matrix are nonnegative, similar conditions are imposed upon bags of size 3 and any set of 3 nodes that is contained in a larger bag.

4.5.2 Bag completion propagation

Although a PSD completion exists for every given bag of 3 nodes, no assignment might exist such that all these bags would satisfy the SDP conditions simultaneously. To find which groups of bags do not have a PSD completion we are applying a propagation algorithm.

After obtaining the solution of the weak relaxation (SDP_R), Theorem 4.5 is applied to those bags of size 3 and 3-node subsets of larger bags where only one line is unassigned. The variables associated with this line are given the unique values satisfying SDP conditions: $w_{ij}^R = \boldsymbol{x}^c$, $w_{ij}^I = \boldsymbol{y}^c$, and the matrix that corresponds to the bag becomes fully defined. After that, the process is repeated, taking the newly calculated values into account.

Algorithm 3 Bag completion propagation

while Number of entries fixed at the previous iteration > 0 **do**
    for Each bag $b \in B(X(\mathbf{x}))$ **do**
        $N_b = \{b_1, b_2, \ldots, b_k\}$, where each b_i is a subset of size 3 of b
        for $i = 1, \ldots, k$ **do**
            if One element (i, j) is undefined in W_{b_i} **then**
                Set $w_{ij}^R = \boldsymbol{x}^c$, $w_{ij}^I = \boldsymbol{y}^c$
            end if
        end for
    end for
end while

Using this algorithm results in finding those negative definite matrices that would otherwise be ignored, thus improving the relaxation gap.

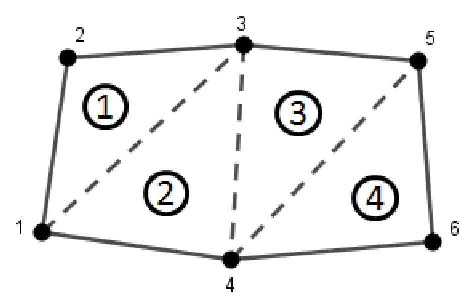


Figure 4.4: A graph with four tree decomposition bags. Existing edges are shown by solid lines, edges added by tree decomposition are shown in dashed lines. Numbers of bags are given in circles.

Let us consider a network with the graph shown in Figure 4.4 as an example. Suppose that tree decomposition yielded four bags: $b_1 = \{1, 2, 3\}$, $b_2 = \{1, 3, 4\}$, $b_3 = \{3, 4, 5\}$ and $b_4 = \{4, 5, 6\}$. After solving the SOCP relaxation, only the values associated with lines that exist in the original graph are known. Algorithm 3 applied to this graph will first consider bag b_1 and assign values to line $(1,3)$. After that bag b_2 will satisfy the conditions of Theorem 4.5, allowing to assign values to line $(3,4)$. Finally, values associated with line $(4,5)$ can be calculated.

After all three dimensional matrices are thus completed, bag b_4 is fully defined and its matrix is not guaranteed to be PSD. Then the SDP condition should be checked for this bag and, if it is violated, a cut should be added to the model.

Which cut will be added depends on the order in which the bags are processed. In our example, if bags are stored in the reverse order, the order in which the lines will be assigned values is reversed too. This can lead to a cut being added for bag b_1 instead of b_4. However, since the cut generation algorithm 2 usually performs several iterations, all relevant cuts are eventually added and the order does not have any significant effect on the results.

4.6 Computational Results

This section presents the results of experiments on 21 small and medium sized instances from the PGLib 17.08 benchmark. Those test cases where the SOCP model gap is greater than 1% were chosen for the experiments. The solver used to solve the NLP problems is Ipopt [187], Mosek [4] was used for SDP problems. The time limit was set to $300s$.

The following models and algorithms were used:

- AC - the nonconvex AC-OPF problem in the polar formulation solved to local optimality by Ipopt;

  Relaxations:

- SOC - the Second Order Cone relaxation of OPF solved by Ipopt;
- FULL - the Semidefinite Programming relaxation of OPF solved by Mosek without exploiting sparsity of the networks;
- SPARSE - the Semidefinite Programming relaxation of OPF solved by Mosek after applying tree decomposition;
- 3D DET - polynomial formulation of SDP-OPF which uses only bags of size 3 and subsets of size 3 of larger bags (see Subsection 4.4.3);
- LIN - the linear cut generation method (see Subsection 4.4.2) with projection subproblems solved by Mosek and the relaxation of OPF solved by Ipopt at each iteration.

In all tables “ERR” and “MEM” respectively indicate that a numerical error has occurred or the solver has run out of memory. The FULL model is the least stable of all, with the solver failing on 16 out of 21 instances. Applying tree decomposition (the SPARSE model) improves stability significantly and decreases the number of failed test cases down to 7. It should be noted that the performance and robustness of SPARSE can be improved [56], however, this was outside the scope of this work. The 3D DET and LIN methods are the most robust: 3D DET fails on one instances and LIN solves all test cases.

Table 4.1: Computational results - gaps

	UB	Gap (%)				
Test case	AC	SOC	FULL	SPARSE	3D DET	LIN
5_pjm	17551.9	5.2	5.2	5.2	5.2	5.6
118_ieee	115804.1	2.1	ERR	**0.2**	**0.2**	0.8
162_ieee_dtc	126154.9	7.7	MEM	**2.4**	**2.4**	2.6
240_pserc	3569993.0	3.8	MEM	**2.3**	**2.3**	**2.3**
300_ieee	664219.6	2.4	MEM	0.4	**0.1**	0.9
3_lmbd__api	11242.1	5.0	5.0	5.0	5.0	5.0
24_ieee_rts__api	134948.2	13.2	ERR	ERR	**2.1**	2.2
30_as__api	4996.2	42.6	ERR	ERR	**7.2**	15.3
30_fsr__api	701.2	2.7	**0.3**	**0.3**	0.8	1.4
39_epri__api	257214.2	1.6	**0.2**	0.3	**0.2**	0.9
73_ieee_rts__api	422726.1	10.8	ERR	ERR	**3.0**	3.1
89_pegase__api	141981.0	8.1	ERR	ERR	**7.0**	7.3
118_ieee__api	316423.5	28.5	ERR	**11.2**	11.8	11.7
162_ieee_dtc__api	143514.5	5.5	MEM	**1.7**	1.8	2.0
24_ieee_rts__sad	76943.2	8.6	ERR	ERR	**2.5**	2.7
57_ieee__sad	45207.6	1.8	**0.1**	**0.1**	0.2	0.8
73_ieee_rts__sad	227745.7	6.7	ERR	ERR	**1.6**	**1.6**
118_ieee__sad	129239.8	11.4	ERR	**3.7**	3.9	4.2
162_ieee_dtc__sad	127038.1	8.3	MEM	**2.4**	ERR	2.7
240_pserc__sad	3656482.5	6.1	MEM	ERR	**4.3**	**4.3**
300_ieee__sad	664309.6	2.3	MEM	0.8	**0.1**	0.9
Average		8.8	2.2	2.6	3.1	3.7

Table 4.1 shows the relative gaps between the upper bounds obtained by solving the AC-OPF model to local optimality and the lower bounds yielded by the convex relaxations, calculated as $\frac{\text{(upper bound)}-\text{(lower bound)}}{\text{(upper bound)}}$. The four strengthened relaxations (FULL, SPARSE, 3D DET and LIN) produce similar or nearly similar (below 1% difference in the gaps) lower bounds in most cases. Since FULL and SPARSE models have equivalent feasible sets, the only difference is in the number of solved instances. The LIN algorithm iterates until the gap is below 1% or all SDP constraints are satisfied, therefore lower bounds should be similar to those yielded by FULL and SPARSE. The computational results confirm this with the exception of instance 30_as__api, where the gaps obtained by solving 3D DET and LIN are 7.2% and 15.3% respectively. On this instance the LIN algorithm terminates early due to numerical issues. An interesting observation can be made regarding the 3D DET formulation: although it ignores matrices of size larger than 3, the resulting gaps are equal to the gaps yielded by FULL and SPARSE.

The strengthened relaxations reduce the gap to below 1% for 6 instances. For many other instances the gap is significantly improved, and the largest differences are observed for instances 30_as__api (35.4 percentage point improvement by 3D DET compared to SOC), 118_ieee__api (17 percentage point improvement) and 24_ieee_rts__api (11 percentage point improvement). No gap reduction has been observed for instances 5_pjm and 3_lmbd__api.

The computational time is presented in Table 4.2. The FULL model is the most computationally expensive, in most cases reaching the time limit or running out of memory. Exploiting sparsity reduces the running time as well as memory requirements dramatically. The 3D DET formulation further improves the performance, being faster than SPARSE on most instances and reducing the average time by 15.7% compared to SPARSE. The LIN algorithm is slower than SPARSE and 3D DET but faster than FULL and is the most robust of the four.

Table 4.2: Computational results - running time (s)

Test case	FULL	SPARSE	3D DET	LIN
5_pjm	**0.0**	0.1	**0.0**	0.5
118_ieee	319.1	1.3	**0.9**	7.8
162_ieee_dtc	MEM	8.5	**5.5**	321.5
240_pserc	MEM	**3.5**	4.4	309.9
300_ieee	MEM	5.4	**2.1**	17.4
3_lmbd__api	**0.0**	**0.0**	**0.0**	0.2
24_ieee_rts__api	1.2	0.3	**0.1**	3.4
30_as__api	4.2	0.3	**0.1**	1.7
30_fsr__api	2.9	0.2	**0.1**	1.9
39_epri__api	11.6	0.3	**0.2**	1.1
73_ieee_rts__api	211.4	1	**0.3**	42.2
89_pegase__api	302.5	**1.7**	8.0	313.8
118_ieee__api	316.9	1.4	**0.9**	37.4
162_ieee_dtc__api	MEM	8.7	**6.9**	323.1
24_ieee_rts__sad	1.3	0.3	**0.1**	3.3
57_ieee__sad	40.9	0.6	**0.2**	0.3
73_ieee_rts__sad	185.2	1	**0.3**	32.4
118_ieee__sad	319.7	1.4	**0.7**	29.8
162_ieee_dtc__sad	MEM	**8.7**	ERR	328.0
240_pserc__sad	MEM	**2.1**	9.1	309.5
300_ieee__sad	MEM	5.4	**2.2**	16.4
Average	122.64	2.49	2.10	100.07

4.7 Conclusion

In this chapter we developed improved cut generation algorithms for SDP problems and applied them to the SDP relaxation of AC-OPF. The notion of the deepest valid cut was defined for separation problems and used to iteratively build a linear programming equivalent of an SDP problem. A polynomial relaxation was introduced based on tree decomposition bags of size 3 and 3-node subsets of larger bags. Computational results show that on selected instances this relaxation yields an optimal objective function value similar to that of the full SDP relaxation of AC-OPF while being more efficient than the standard sparse SDP formulation. The dynamic linearisation algorithm, although slower than the static sparse models, provides the best robustness and has the potential to be used for mixed-integer semidefinite programming.

Chapter 5

Convex Hulls for Quadratic On/Off Constraints

5.1 Introduction

Continuous models studied in Chapters 3 and 4 can represent only a limited subset of real life applications. The need to apply optimisation methods to systems containing discrete control elements motivates studies on mixed-integer programming, where some variables are required to be integer or binary. Such control elements often function by switching the system between two states, the "on" and the "off" state, and these are modelled by on/off or indicator constraints. These constraints have the form of a continuous constraint which is activated or deactivated depending on the value of some binary variable.

In this chapter we propose a convex hull formulation for a two-dimensional on/off constraint defined by a nonmonotone quadratic function. This characterisation does not use any additional variables and extends a previously proved result [98] which required the functions to be coordinate-wise monotone. Then we study trigonometric constraints and introduce their tight convex quadratic relaxations, which can be solved by efficient quadratic programming solvers.

These theoretical results are applied to the Quadratic Convex (QC) relaxation of the Alternating Current Optimal Transmission Switching (OTS) problem. To further strengthen the QC-OTS model, we tighten the big-M constraints by finding better values of the constants, add valid cuts and apply bound propagation. The experiments show that using the new convex hull formulation improves the average performance and the tightened relaxation results tighter lower bounds, with the best bound improvement yielded by bound propagation.

The rest of the chapter is organised as follows. Section 5.2 gives an overview of mixed-integer nonlinear programming methods. In Section 5.3 we discuss the approach known as disjunctive programming, give the definition of a perspective function and review the literature on constructing convex formulations of disjunctive sets. In Section 5.4 we provide the proof for our new convex hull characterization. Quadratic relaxations of trigonometric functions are derived in Section 5.5. Section 5.6 focuses on tightening the QC relaxation of the OTS problem, using the results of the previous sections. Section 5.7 reports the computational results and Section 5.8 concludes the chapter.

5.2 Mixed-Integer Nonlinear Programming

Consider a Mixed-Integer Nonlinear Program

$$\begin{aligned} \min \;& f(\mathbf{x},\mathbf{y}) \\ \text{s.t. }& g_i(\mathbf{x},\mathbf{y},\mathbf{z}) \leqslant 0, \;\; \forall i = 1,\dots,m, \\ & \mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{Z}^{n_I}, \mathbf{z} \in \{0,1\}^{n_B}, \end{aligned} \tag{MINLP}$$

where functions f and g_i, $i = 1,\dots,m$ are assumed to be continuous and twice differentiable.

MINLPs extend the applicability of mathematical optimisation to systems described by nonlinear constraints where discrete control decisions are made. These are captured by the integer ($\mathbf{y} \in \mathbb{Z}^{n_I}$) and binary variables ($\mathbf{z} \in \{0,1\}^{n_B}$) and present a computational challenge because of the combinatorial nature of the problem. The integrality requirements can also be viewed as a source of non-convexity.

Significant progress has been made in the area of mixed-integer linear programming (MILP), where all functions f and g_i, $i = 1,\dots,m$ are linear. Its development began with two papers that proposed using linear programming (LP) and cutting planes for solving MILPs [51, 81]. These methods start with LP relaxations obtained by discarding the integrality requirements and then iteratively cut off integer infeasible solutions. This was followed by the introduction of the branch and bound method [118, 50]. Similar to the cutting planes method, it first solves an LP relaxation. After that, if the solution violates the integrality requirements, the branch and bound algorithm chooses one integer variable that has a fractional value in the current solution and splits its domain in two (this splitting is referred to as branching). Subproblems corresponding to the two resulting subregions are solved, and then, if needed, branching is performed again. The branch and cut method [11] combines the branch and bound and cutting plane algorithms, and the branch and price method [13, 168] incorporates column generation.

A particular challenge is presented by MINLPs which combine the difficulties posed by discrete decisions and (possibly nonconvex) nonlinearities. The nonlinear programming (NLP) based branch and bound method, which is similar in idea to branch and bound for MILPs but uses nonlinear continuous relaxations, was first mentioned by Dakin [50] and later developed by Gupta and Ravindran [88]. The main obstacle in applying this method to large MINLPs is that solving a large number of NLP subproblems can be very computationally expensive. LP-based branch and bound [179] for MINLPs constructs polyhedral relaxations of the node problems, enabling the use of LP methods. These algorithms are enhanced by presolving, heuristics that can find better feasible solutions earlier in the search process, cut generation, smart branching rules, tightening continuous relaxations, column generation and other techniques.

The idea of iteratively building MILP relaxations of the feasible set of a MINLP is implemented in the outer approximation and generalised Benders decomposition algorithms. Both approaches divide the problem variables into "complicating" integer and "easy" continuous ones and solve subproblems obtained by fixing the integer variables. The outer approximation method [54, 61] solves a sequence of MILPs obtained by linearising the constraints around a subset of feasible solutions. Points are iteratively added to this subset, thus refining the MILP relaxation of the MINLP. Generalised Benders decomposition [73], which extends the algorithm of Benders [20] to the case of MINLPs, is very similar to outer approximation but uses dual information to obtain the linearisations.

For an in-depth review of the methods used for solving MINLPs we refer the reader to the surveys [24, 85].

5.2.1 Branch and bound

Branch and bound is a methodology of systematic search which is done by analysing subregions of the problem. Here we will describe the nonlinear programming based branch and bound algorithm.

The search process can be viewed as exploring a tree with each node corresponding to a particular subregion (an example of such a tree is given by Figure 5.1). The algorithm keeps track of two key values. One is the lower bound obtained by choosing the smallest objective value of continuous NLP relaxations of all the current leaf nodes. These are obtained by disregarding the integrality requirements in (MINLP):

$$\begin{aligned} \min\ & f(\mathbf{x},\mathbf{y}) \\ \text{s.t. } & g_i(\mathbf{x},\mathbf{y},\mathbf{z}) \leqslant 0,\ \forall i \in 1,\dots,m, \\ & \mathbf{z}_j = \hat{\mathbf{z}}_j,\ \forall j \in I^z, \\ & \mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^{n_I}, \\ & \mathbf{z} \in \mathbb{R}^{n_B}, \hat{\mathbf{z}} \in \{0,1\}^{|I^z|}, \end{aligned} \tag{NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)}$$

where $(\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z)$ depend on the branching decisions. $\mathbf{y}^L$ and $\mathbf{y}^U$ denote upper and lower bounds on $\mathbf{y}$, and $\hat{\mathbf{z}}$ is a constant vector with indices in I^z.

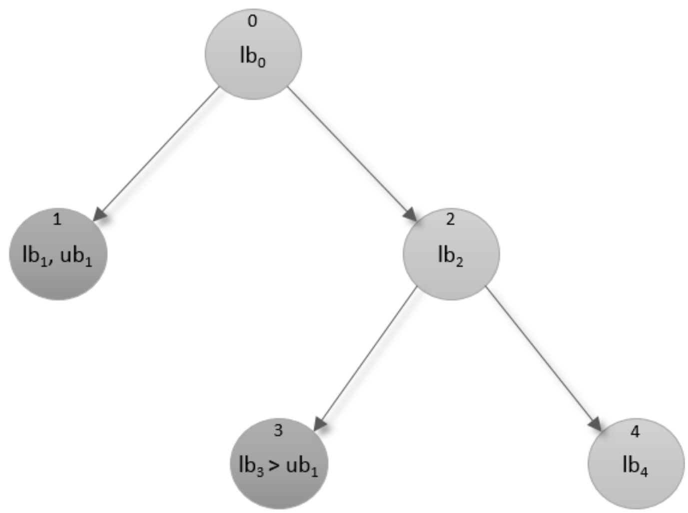
For the example shown in Figure 5.1, the current lower bound is equal to $\min(lb_1, lb_3, lb_4)$.

The other value is the upper bound, which is the objective value yielded by the best known feasible solution, also called incumbent solution (ub_1 in Figure 5.1).

The algorithm follows the following steps:

- Root node (node 0 in Figure 5.1).

Figure 5.1: A branch and bound tree. The fathomed nodes are marked by darker grey colour.



The algorithm starts at the root node where no branching has been done yet. The continuous NLP relaxation (NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)) is obtained by dropping the requirements $\mathbf{y} \in \mathbb{Z}^{n_I}$, $\mathbf{z} \in \{0,1\}^{n_B}$ from (MINLP). The resulting nonlinear program is solved to find the first lower bound on the optimal solution. If all discrete variables satisfy the integrality requirement at the solution, terminate since we have found the optimal solution of the original problem (which happens very rarely in practice).

- Branching.

  A discrete variable is chosen and its domain is divided into two parts. This produces two new subproblems (which can be represented as two new nodes in the search tree) with additional constraints on the discrete variables.

- Processing a leaf node.

  Solve (NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)), which is the continuous NLP relaxation of a subproblem, and analyse the solution:

  – If (NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)) at the node is infeasible, then this node as well as any of its children nodes do not contain any feasible solutions. The node is designated as fathomed: there is no need to further explore the tree in this direction.

  – If the lower bound at the node is higher than the current upper bound, it means that further exploring the tree by branching from this node will not yield any improved feasible solutions. The node is designated as fathomed. (Node 3 in Figure 5.1)

  – If the solution of (NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)) satisfies the integrality requirements and is better than the current incumbent, we have found a new best integer feasible solution. Update the incumbent. There is no need to further branch from this node. (Node 1 in Figure 5.1)

– Otherwise, branch again from this node.

As the search thus progresses, the upper and lower bounds are updated until their relative difference, referred to as the optimality gap, becomes smaller than a given tolerance.

The role of continuous relaxations At each node of the branch and bound tree, an NLP relaxation (NLP($\mathbf{y}^L, \mathbf{y}^U, \hat{\mathbf{z}}, I^z$)) of the corresponding subproblem is solved and the result is used to update the lower bound and determine which nodes should be fathomed. The tighter the relaxation is, the more information can be derived from solving it.

The quality of the NLP relaxation depends on the formulation of the problem. Characterisations that describe the same mixed-integer set might result in different continuous relaxations. The number of variables and constraints affects the efficiency as well. Therefore finding better problem formulations can lead to improved performance.

5.3 On/Off Constraints

On/off constraints have the form,

$$g(\mathbf{x}) \leqslant 0 \text{ if } z = 1, \tag{5.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and z is a binary variable. In this work, function g will be assumed to be continuous, twice differentiable and convex.

(5.1) is also known as a disjunctive or indicator constraint. We assume that the variable bounds are part of the disjunction, i.e.,

$$\mathbf{x}^{l0} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u0}, \text{ if } z = 0, \tag{5.2}$$

$$\mathbf{x}^{l1} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u1}, \text{ if } z = 1. \tag{5.3}$$

The question is how to formulate constraints (5.1)-(5.3) algebraically, so that a problem that contains disjunctions can be written in the general form (MINLP) and solved by MINLP solvers. When constructing the formulation, we need to take into account the following considerations:

- the continuous relaxation should be convex,
- using fewer variables and constraints is preferable,
- tighter continuous relaxations lead to better performance.

Big-M relaxations [143] are commonly used for constructing algebraic formulations of on/off constraints. For a constraint in the form (5.1), the big-M relaxation is written as $g(\mathbf{x}) \leqslant \boldsymbol{M}(1 - z)$, where $\boldsymbol{M}$ is a constant number large enough so that the constraint becomes redundant if $z = 0$. Such formulations do not employ additional variables and are convex, but often lead to weak continuous relaxations.

Alternatively, these constraints can be viewed from a disjunctive programming standpoint. Disjunctive programming deals with problems expressed by algebraic constraints and logic disjunctions. The advantage of this representation is that it naturally reflects the logic structure of the problem and is helpful in building better relaxations. Applying disjunctive programming to mixed-integer problems was first suggested by Jeroslow and Lowe [104] and Balas [9, 10] in the context of MILP. Generalised disjunctive programming [160] is an extension of disjunctive programming that allows more complex logical relations.

With this approach, an on/off constraint is described by a union of two sets, with each set corresponding to a value of the binary variable. Constraints (5.1)-(5.3) can thus be reformulated as:

$$\begin{gathered}(\mathbf{x}, z) \in \Gamma_0 \cup \Gamma_1, \\ \Gamma_0 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 0,\ \mathbf{x}^{l_0} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_0}\}, \\ \Gamma_1 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 1,\ g(\mathbf{x}) \leqslant 0,\ \mathbf{x}^{l_1} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_1}\}.\end{gathered} \tag{5.4}$$

Let us introduce the notion of a convex hull which proves to be useful in this context.

Definition 5.1. *Given a set S, its convex hull conv(S) is the smallest convex set that contains S.*

The set described by (5.4) can be equivalently characterised by requiring that $(\mathbf{x}, z)$ belong to the convex hull of two sets and z is binary:

$$\begin{gathered}(\mathbf{x}, z) \in \operatorname{conv}(\Gamma_0 \cup \Gamma_1), \\ z \in \{0,1\}.\end{gathered} \tag{5.5}$$

Dropping the integrality requirement on variable z results in a convex continuous relaxation of (5.4) which is typically tighter than the big-M relaxation. Moreover, since, by definition, a convex hull is the smallest convex set containing a given set, it is the tightest possible convex relaxation of $\Gamma_0 \cup \Gamma_1$. The challenging task lies in finding a compact algebraic characterization of set (5.5), i.e., a representation defined in the space of original variables.

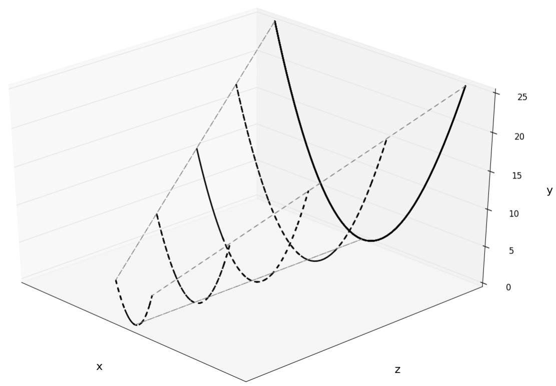
5.3.1 Perspective functions

Before we proceed to reviewing the existing methods for constructing convex hull formulations, let us describe a useful tool known as the perspective function.

Definition 5.2. *(Perspective function) [167] For a given convex function $f : \mathbb{R}^n \to \mathbb{R}$, its perspective function $\tilde{f} : \mathbb{R}^{n+1} \to (\mathbb{R} \cup \{+\infty\})$ is defined as:*

$$\tilde{f}(\mathbf{x}, z) = \begin{cases} z f(\mathbf{x}/z) & \text{if } z > 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Figure 5.2: Several dilations of the function $y = x^2$



For each fixed $z = \boldsymbol{z}^0$ the function $\tilde{f}(\mathbf{x}, \boldsymbol{z}^0)$ represents a dilation of the original function $f(\mathbf{x})$.

A perspective function has a focal point, which is a point approached by the dilations as z approaches 0. By modifying the argument of the perspective function one can modify its focal point.

Importantly, the perspective operator preserves convexity, i.e., if function f is convex, so will be its perspective $\tilde{f}$ [103].

These properties are used to build convex hulls.

5.3.2 Formulating the convex hull

Extensive work has been done on deriving convex hulls of unions of sets and applying these results to enhance mixed-integer programming formulations and algorithms.

Jeroslow [103] proved that the perspective operator preserves convexity and suggested using it for formulating convex hulls of unions of convex sets. Using a similar convexification technique, Stubbs and Mehrotra [178] generalised the MILP branch and cut algorithm [11] to the case of convex problems that involve binary variables. These works inspired more research on improving MINLP modelling and solution techniques with convex hull formulations [34, 86, 2].

However, all these works employ a convexification approach that requires adding auxiliary variables to the original formulation, thus increasing the model size and decreasing its computational efficiency.

Based on the perspective formulation, linear cuts expressed in the space of original variables were derived and used in a cut generating approach proposed by Frangioni et al. [63, 64]. Günlük and Linderoth [87] considered on/off constraints where the set Γ_0 reduces to a single point and proposed the convex hull characterisation without adding new variables.

Hijazi et al. [98] generalised this result to cases where Γ_0 is a hyper-rectangle and the constraints are isotone (coordinate-wise monotone). In a recent work, Belotti et al. [17]

study the efficiency of nonconvex formulations for on/off constraints in conjunction with aggressive bound tightening techniques.

Convex hulls of isotone functions We will restate a result presented by Hijazi et al. [98], which characterises the convex hull of a union of two convex sets defined by isotone functions.

Isotone functions are one of possible extensions of the notion of monotone functions to multiple dimensions and are characterised by being coordinate-wise monotone:

Definition 5.3. [98] Let $f : E \to \mathbb{R}, E \subseteq \mathbb{R}^n$.

- f is *independently increasing* (resp. *decreasing*) on coordinate i if for all $\mathbf{x} \in \text{dom}(f)$ and $\lambda > 0$ such that $\mathbf{x} + \lambda e_i \in dom(f)$, where e_i is ith unit vector of the standard basis, we have $f(\mathbf{x} + \lambda e_i) \geqslant f(\mathbf{x})$ (resp. $f(\mathbf{x} + \lambda e_i) \leqslant f(\mathbf{x})$).
- f is *independently monotone* on coordinate i if it is independently increasing or independently decreasing on the ith coordinate.
- f is *isotone* if it is independently monotone on every coordinate.

Under the assumption of function isotonicity, the following theorem provides the formulation of a convex hull:

Theorem 5.1. [98] Let $f : E \to \mathbb{R}, E \subseteq \mathbb{R}^n$, be an isotone closed convex function with J^1 (resp., J^2) the set of indices on which f is independently increasing (resp. decreasing),

$$\Gamma_0 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 0,\ \mathbf{x}^{l_0} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_0}\},$$
$$\Gamma_1 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 1,\ f(\mathbf{x}) \leqslant 0,\ \mathbf{x}^{l_1} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_1}\} \neq \varnothing,$$

then $\text{conv}(\Gamma_0 \cup \Gamma_1) = \text{closure}(\Gamma')$, where

$$\Gamma' = \left\{ \begin{array}{l} (\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid \\ z q_S(\mathbf{x}, z) \leqslant 0 \quad \forall S \subset \{1, 2, \ldots, n\} \\ z\mathbf{x}^{l_1} + (1-z)\mathbf{x}^{l_0} \leqslant x \leqslant z\mathbf{x}^{u_1} + (1-z)\mathbf{x}^{u_0} \\ 0 < z \leqslant 1 \end{array} \right\},$$

$q_S = (f \circ h_S)$ and $h_S(\mathbb{R}^n \times [0,1] \to \mathbb{R}^n)$ is defined by

$$(h_S(\mathbf{x}, z))_i = \begin{cases} \boldsymbol{x}_i^{l_1} & \forall i \in S \cap J_1, \\ \boldsymbol{x}_i^{u_1} & \forall i \in S \cap J_2, \\ \frac{x_i - (1-z)\boldsymbol{x}_i^{u_0}}{z} & \forall i \in J_1,\ i \notin S, \\ \frac{x_i - (1-z)\boldsymbol{x}_i^{l_0}}{z} & \forall i \in J_2,\ i \notin S. \end{cases}$$

Although this formulation is given in the space of original variables, it involves an exponential number of constraints since in the definition of Γ', an inequality has to be added for each possible combination of index values $1, \ldots, n$. A relaxation involving a subset of constraints defined above has been proposed:

Corollary 5.1. *[98] Let $f : E \to \mathbb{R}, E \subseteq \mathbb{R}^n$, be an isotone closed convex function with J^1 (resp., J^2) the set of indices on which f is independently increasing (resp. decreasing),*

$$\Gamma_0 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 0,\ \mathbf{x}^{l_0} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_0}\},$$
$$\Gamma_1 = \{(\mathbf{x}, z) \in \mathbb{R}^n \times \{0,1\} \mid z = 1,\ f(\mathbf{x}) \leqslant 0,\ \mathbf{x}^{l_1} \leqslant \mathbf{x} \leqslant \mathbf{x}^{u_1}\} \neq \varnothing,$$

$$\Gamma'' = \left\{ \begin{array}{l} (\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid \\ z q_{\varnothing}(\mathbf{x}, z) \leqslant 0 \\ z\mathbf{x}^{l_1} + (1-z)\mathbf{x}^{l_0} \leqslant x \leqslant z\mathbf{x}^{u_1} + (1-z)\mathbf{x}^{u_0} \\ 0 < z \leqslant 1 \end{array} \right\},$$

with $q_{\varnothing} = (f \circ h_S)$ and $h_{\varnothing}(\mathbb{R}^n \times [0,1] \to \mathbb{R}^n)$ defined by

$$(h_{\varnothing}(\mathbf{x}, z))_i = \begin{cases} \frac{x_i - (1-z)\boldsymbol{x}_i^{u_0}}{z} & \forall i \in J_1, \\ \frac{x_i - (1-z)\boldsymbol{x}_i^{l_0}}{z} & \forall i \in J_2. \end{cases}$$

For a linear constraint where $g(\mathbf{x}) = \boldsymbol{a}^T\mathbf{x} - b$, the perspective-based relaxation is written as:

$$\Gamma'' = \left\{ \begin{array}{l} (\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid \\ \sum\limits_{i \in N} \boldsymbol{a}_i x_i \leqslant z \left(b - \sum\limits_{\substack{i \in N \\ \boldsymbol{a}_i < 0}} \boldsymbol{a}_i \boldsymbol{x}_i^{l_0} - \sum\limits_{\substack{i \in N \\ \boldsymbol{a}_i > 0}} \boldsymbol{a}_i \boldsymbol{x}_i^{u_0} \right) \\ 0 \leqslant z \leqslant 1 \end{array} \right\} [101]. \tag{5.6}$$

In this chapter we are extending the result of Theorem 5.1 to non-isotone functions for the quadratic two-dimensional case:

$$\boldsymbol{a}x^2 + \boldsymbol{b}x + \boldsymbol{c} - y \leqslant 0, \text{ if } z = 1\ (\boldsymbol{a} > 0).$$

5.4 Convex Hull of a Nonmonotone Quadratic Constraint

We start by recalling the following known result about convex hulls.

Lemma 5.1. *[25] Let $D = D_1 \cup D_2$,*

$$\textit{then } \mathsf{conv}(D) = \mathsf{conv}(\mathsf{conv}(D_1) \cup \mathsf{conv}(D_2)).$$

This Lemma allows us to first find convex hulls of two separate sets and then use them in the construction of the convex hull of their union.

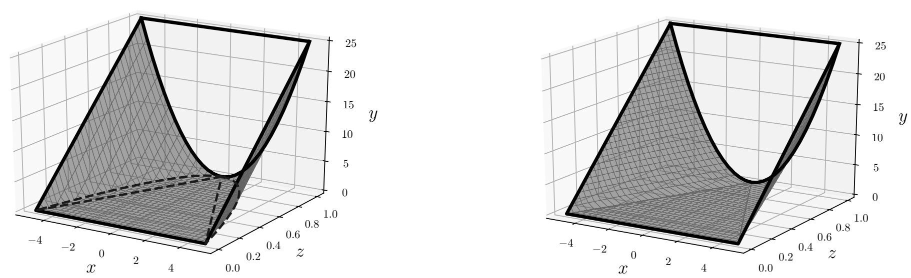
Now, we shall prove the main result of this Chapter.

Theorem 5.2. *Let* $f(x,y) = \boldsymbol{a}x^2 + \boldsymbol{b}x + \boldsymbol{c} - y,\ \boldsymbol{a} > 0,$
$\Gamma_0 = \left\{(x,y,z) \in \mathbb{R}^2 \times \mathbb{B} \mid z = 0,\ \boldsymbol{x}^{l_0} \leqslant x \leqslant \boldsymbol{x}^{u_0},\ y = 0\right\}$, *and*
$\Gamma_1 = \left\{(x,y,z) \in \mathbb{R}^2 \times \mathbb{B} \;\middle|\; z = 1,\ \boldsymbol{x}^{l_1} \leqslant x \leqslant \boldsymbol{x}^{u_1},\ \boldsymbol{y}^{l} \leqslant y \leqslant \boldsymbol{y}^{u},\ f(x,y) \leqslant 0\right\}$

then $\mathsf{conv}(\Gamma_0 \cup \Gamma_1) =$

$$\left\{ (x,y,z) \in \mathbb{R}^2 \times [0,1] \;\middle|\; \begin{array}{l} x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{yz + \boldsymbol{\delta} z^2}{\boldsymbol{a}}}, \\[2ex] x - \boldsymbol{x}^{l_0}(1-z) + \boldsymbol{\rho} z \geqslant -\sqrt{\frac{yz + \boldsymbol{\delta} z^2}{\boldsymbol{a}}}, \\[2ex] z\boldsymbol{x}^{l_1} + (1-z)\boldsymbol{x}^{l_0} \leqslant x \leqslant z\boldsymbol{x}^{u_1} + (1-z)\boldsymbol{x}^{u_0}, \\[2ex] \boldsymbol{y}^{l} z \leqslant y \leqslant \boldsymbol{y}^{u} z, \end{array} \right\}$$

where $\boldsymbol{\rho} = \frac{\boldsymbol{b}}{2\boldsymbol{a}}$ *and* $\boldsymbol{\delta} = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c}$.



Big-M formulation   Convex hull formulation

Figure 5.3: Tightening convex relaxations

Proof. First, we split Γ_1 into

$$\Gamma_1^r = \left\{(x,y,z) \in \Gamma_1 \mid -\boldsymbol{\rho} \leqslant x \leqslant \boldsymbol{x}^{u_1}\right\}, \text{ and } \Gamma_1^l = \left\{(x,y,z) \in \Gamma_1 \mid \boldsymbol{x}^{l_1} \leqslant x \leqslant -\boldsymbol{\rho}\right\}.$$

Consider the set $\Gamma^r = \Gamma_0 \cup \Gamma_1^r$. For $x \in \Gamma_1^r$, $f(x,y)$ is isotone, and its inverse can be taken. The inequality $f(x,y) \leqslant 0$ can be rewritten as:

$$f^r(x,y) = x + \boldsymbol{\rho} - \sqrt{\frac{y + \boldsymbol{\delta}}{\boldsymbol{a}}} \leqslant 0$$

$f^r(x,y)$ is isotone everywhere in its domain, thus Theorem 5.1 can be applied. Let us first construct the functions zq_S.

Let $\boldsymbol{x}_r^{l_1} = \mathsf{max}(-\boldsymbol{\rho}, \boldsymbol{x}^{l_1})$.

• $[S = \varnothing]$ $h_{\varnothing}(x,y,z) = \begin{pmatrix} (x - \boldsymbol{x}^{u_0}(1-z))/z \\ y/z \end{pmatrix}$,

$$zq_{\varnothing} = zf(h_{\varnothing}(x,y,z)) = x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z - \sqrt{\frac{yz + \boldsymbol{\delta} z^2}{\boldsymbol{a}}},$$

• $[S = \{1\}]$ $h_1(x,y,z) = \begin{pmatrix} \boldsymbol{x}_r^{l_1} \\ y/z \end{pmatrix}$,

$$zq_1 = zf(h_1(x,y,z)) = (\boldsymbol{x}_r^{l_1} + \boldsymbol{\rho})z - \sqrt{\frac{yz + \boldsymbol{\delta} z^2}{\boldsymbol{a}}},$$

• $[S = \{2\}]$ $h_2(x,y,z) = \begin{pmatrix} (x - (1-z)\boldsymbol{x}^{u_0})/z \\ \boldsymbol{y}^u \end{pmatrix}$,

$$zq_2 = zf(h_2(x,y,z)) = x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z - z\sqrt{\frac{\boldsymbol{y}^u + \boldsymbol{\delta}}{\boldsymbol{a}}}.$$

The convex hull of Γ^r is then given by:

$$\mathsf{conv}(\Gamma^r) =$$

$$\left\{ (x,y,z) \in \mathbb{R}^2 \times [0,1] \;\middle|\; \begin{array}{l} x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}}, \\ (\boldsymbol{x}_r^l + \boldsymbol{\rho})z \leqslant \sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}} \\ x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z \leqslant z\sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} \\ \boldsymbol{x}_r^{l_1} z + \boldsymbol{x}^{l_0}(1-z) \leqslant x \leqslant \boldsymbol{x}^{u_1} z + \boldsymbol{x}^{u_0}(1-z), \\ \boldsymbol{y}^l z \leqslant y \leqslant \boldsymbol{y}^u z. \end{array} \right\}$$

For $x \in \Gamma^l$, the inequality $f(x,y) \leqslant 0$ is equivalent to:

$$f^l(x,y) = -x - \boldsymbol{\rho} - \sqrt{\frac{y + \boldsymbol{\delta}}{\boldsymbol{a}}} \leqslant 0.$$

The convex hull of $\Gamma^l = \Gamma_0 \cup \Gamma_1^l$ can be obtained similarly:

$$\mathsf{conv}(\Gamma^l) =$$

$$
\left\{ (x,y,z) \in \mathbb{R}^2 \times [0,1] \;\middle|\; \begin{array}{l} x - \boldsymbol{x}^{l_0}(1-z) + \boldsymbol{\rho} z \geqslant -\sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}}, \\ (\boldsymbol{x}_l^u + \boldsymbol{\rho})z \geqslant -\sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}} \\ x - \boldsymbol{x}^{l_0}(1-z) + \boldsymbol{\rho} z \geqslant -z\sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} \\ \boldsymbol{x}^{l_1} z + \boldsymbol{x}^{l_0}(1-z) \leqslant x \leqslant \boldsymbol{x}_l^{u_1} z + \boldsymbol{x}^{u_0}(1-z), \\ \boldsymbol{y}^l z \leqslant y \leqslant \boldsymbol{y}^u z, \end{array} \right\}
$$

where $\boldsymbol{x}_l^{u_1} = \mathsf{min}(-\boldsymbol{\rho}, \boldsymbol{x}^{u_1})$.

Redundancy of constraints.

1. The first constraint in the formulation of $\mathsf{conv}(\Gamma^r)$ can be nonredundant in Γ^l only if the solution space of the following system is nonempty:

$$
\begin{cases} x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z > 0 \\ x \leqslant \boldsymbol{x}_l^{u_1} z + \boldsymbol{x}^{u_0}(1-z). \end{cases}
$$

   Solutions for this system exist if and only if the following inequality has solutions:

$$
\boldsymbol{x}_l^{u_1} z + \boldsymbol{x}^{u_0}(1-z) > \boldsymbol{x}^{u_0}(1-z) - \boldsymbol{\rho} z.
$$

   Since $\boldsymbol{x}_l^{u_1} \leqslant -\boldsymbol{\rho}$, no $z \leqslant 1$ satisfying this inequality exists and the first constraint in $\mathsf{conv}(\Gamma^r)$ is redundant in Γ^l. Similarly, the first constraint in the formulation of $\mathsf{conv}(\Gamma^l)$ is redundant in Γ^r.

2. Consider the constraint $(\boldsymbol{x}_r^{l_1} + \boldsymbol{\rho})z \leqslant \sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}}$. If $\boldsymbol{x}_r^{l_1} = -\boldsymbol{\rho}$, then the left hand side is zero and the inequality is always satisfied. Now suppose that $\boldsymbol{x}_r^{l_1} = \boldsymbol{x}^{l_1} > -\boldsymbol{\rho}$. In this case, we have that $(\boldsymbol{x}_r^{l_1} + \boldsymbol{\rho})z \geqslant 0$, and the constraint can be rewritten as:

$$
(\boldsymbol{a}(\boldsymbol{x}^{l_1} + \boldsymbol{\rho})^2 - \boldsymbol{\delta})z^2 \leqslant yz \;\Leftrightarrow (\boldsymbol{a}(\boldsymbol{x}^{l_1})^2 + \boldsymbol{b}\boldsymbol{x}^{l_1} + \boldsymbol{c})z \leqslant y.
$$

   Recalling that for $x > -\boldsymbol{\rho}$ the function $\boldsymbol{a}x^2 + \boldsymbol{b}x + \boldsymbol{c}$ increases and $\boldsymbol{a}x^2 + \boldsymbol{b}x + \boldsymbol{c} \leqslant y \;\forall (x,y) \in \Gamma_1$, it is safe to assume that $\boldsymbol{y}^l \geqslant \boldsymbol{a}(\boldsymbol{x}^{l_1})^2 + \boldsymbol{b}\boldsymbol{x}^{l_1} + \boldsymbol{c}$. Therefore, constraint $(\boldsymbol{x}_r^{l_1} + \boldsymbol{\rho})z \leqslant \sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}}$ is dominated by $\boldsymbol{y}^l z \leqslant y$. Similarly, constraint $(\boldsymbol{x}_l^{u_1} + \boldsymbol{\rho})z \geqslant -\sqrt{\frac{yz+\boldsymbol{\delta} z^2}{\boldsymbol{a}}}$ is either always satisfied or dominated by $\boldsymbol{y}^l z \leqslant y$.

3. Consider the constraint $x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho} z \leqslant z\sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}}$. Rewrite it as:

$$
x \leqslant \left(\sqrt{\frac{\boldsymbol{y}^u + \boldsymbol{\delta}}{\boldsymbol{a}}} - \boldsymbol{\rho} \right) z + \boldsymbol{x}^{u_0}(1-z).
$$

Observe that $x = \sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} - \boldsymbol{\rho}$ is the solution of equation $f^r(x, \boldsymbol{y}^u) = 0$. Since $\boldsymbol{a}x^2+\boldsymbol{b}x+\boldsymbol{c}$ increases for $x > -\rho$, all points (x, y) such that $x > \sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} - \boldsymbol{\rho}$ and $f(x, y) \leqslant 0$ violate the upper bound on y. Then we can assume without loss of generality that $\boldsymbol{x}^{u_1} \leqslant \sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} - \boldsymbol{\rho}$. This implies that the constraint $x \leqslant \left(\sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}} - \boldsymbol{\rho}\right) z + \boldsymbol{x}^{u_0}(1-z)$ is dominated by $x \leqslant z\boldsymbol{x}^{u_1} + (1-z)\boldsymbol{x}^{u_0}$. Similarly, constraint $x - \boldsymbol{x}^{l_0}(1-z) + \boldsymbol{\rho}z \geqslant -z\sqrt{\frac{\boldsymbol{y}^u+\boldsymbol{\delta}}{\boldsymbol{a}}}$ is dominated by $x \geqslant z\boldsymbol{x}^{l_1} + (1-z)\boldsymbol{x}^{l_0}$.

The full convex hull. By Lemma 5.1 we have

$$\mathsf{conv}(\Gamma^r \cup \Gamma^l) = \mathsf{conv}(\mathsf{conv}(\Gamma^r) \cup \mathsf{conv}(\Gamma^l)).$$

This and the fact that $\Gamma_0 \cup \Gamma_1 = \Gamma^r \cup \Gamma^l$ allow us to construct $\mathsf{conv}(\Gamma_0 \cup \Gamma_1)$ by taking a union of the two sets defined above:

$$\mathsf{conv}(\Gamma_0 \cup \Gamma_1) = \mathsf{conv}(\mathsf{conv}(\Gamma^r) \cup \mathsf{conv}(\Gamma^l)) =$$

$$\left\{ (x, y, z) \in \mathbb{R}^2 \times [0, 1] \;\middle|\; \begin{aligned} & x - \boldsymbol{x}^{u_0}(1-z) + \boldsymbol{\rho}z \leqslant \sqrt{\tfrac{yz+\boldsymbol{\delta}z^2}{\boldsymbol{a}}} \\ & x - \boldsymbol{x}^{l_0}(1-z) + \boldsymbol{\rho}z \geqslant -\sqrt{\tfrac{yz+\boldsymbol{\delta}z^2}{\boldsymbol{a}}} \\ & z\boldsymbol{x}^{l_1} + (1-z)\boldsymbol{x}^{l_0} \leqslant x \leqslant z\boldsymbol{x}^{u_1} + (1-z)\boldsymbol{x}^{u_0} \\ & \boldsymbol{y}^l z \leqslant y \leqslant \boldsymbol{y}^u z. \end{aligned} \right\}$$

□

Figure 5.3 compares the convex hull to the region defined by the big-M constraint. The meshed surface represents the boundary of the feasible set of the corresponding constraint, and the interior of feasible set consists of all points above the surface.

If a solver that does not support nonlinear nonquadratic constraints is used, the above formulation can be utilised in order to generate linear cuts. An example of such use is shown in Subsection 5.6.3.

5.5 Quadratic Outer Approximations of Trigonometric Functions

In this section, we derive quadratic relaxations for trigonometric functions $f(x)$, $\boldsymbol{x}^l \leqslant x \leqslant \boldsymbol{x}^u$, and we consider the case $(\boldsymbol{x}^u - \boldsymbol{x}^l) < \pi/2$, with asymmetrical bounds. The choice of $\boldsymbol{x}^l$ and $\boldsymbol{x}^u$ should ensure that $f(x)$ is either convex or concave on the interval $[\boldsymbol{x}^l, \boldsymbol{x}^u]$. To the best of our knowledge, this is the first quadratic relaxation of trigonometric functions exploiting asymmetrical bounds on x.

Let $Q_f(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$ denote the quadratic function passing through three distinct points

$(\boldsymbol{x}_1; f(\boldsymbol{x}_1)), (\boldsymbol{x}_2; f(\boldsymbol{x}_2))$, and $(\boldsymbol{x}_3; f(\boldsymbol{x}_3))$:

$$Q_f(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\boldsymbol{\delta}_{32}}{\boldsymbol{\delta}_{21}\boldsymbol{\delta}_{31}\boldsymbol{\delta}_{32}}(x - \boldsymbol{x}_1)(x - \boldsymbol{x}_2) + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}}(x - \boldsymbol{x}_2) + f(\boldsymbol{x}_2),$$

where $\boldsymbol{\delta}_{ij} = \boldsymbol{x}_i - \boldsymbol{x}_j$ and $\boldsymbol{\phi}_{ij} = f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)$.

We will prove that with the correct choice of $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$, Q_f overestimates f on $[\boldsymbol{x}^l, \boldsymbol{x}^u]$ and thus can be used in building a valid relaxation of a set described by $f(x) = 0$, $x \in [\boldsymbol{x}^l, \boldsymbol{x}^u]$ or $f(x) \leqslant 0$, $x \in [\boldsymbol{x}^l, \boldsymbol{x}^u]$.

Proposition 5.1. *Given* ϵ *s.t.* $0 < \epsilon < \frac{\pi}{2} - \boldsymbol{x}^u$*, if* $0 \leqslant \boldsymbol{x}^l \leqslant \boldsymbol{x}^u < \frac{\pi}{2}$*, then*

$$\cos(\boldsymbol{x}^u + \epsilon) \leqslant \cos(\boldsymbol{x}^u) - \epsilon \sin(\boldsymbol{x}^u).$$

Proof. Consider the tangent to the function $\cos(x)$ at $x = \boldsymbol{x}^u$. Its equation is written as $f(x) = \cos(\boldsymbol{x}^u) - \sin(\boldsymbol{x}^u)(x - \boldsymbol{x}^u)$. It lies above the cosine function since $\cos(x)$ is concave for $0 < x < \frac{\pi}{2}$.

Then for all $0 \leqslant \epsilon \leqslant \frac{\pi}{2} - \boldsymbol{x}^u$ we have:

$$\cos(\boldsymbol{x}^u + \epsilon) \leqslant f(\boldsymbol{x}^u + \epsilon) = \cos(\boldsymbol{x}^u) - \epsilon \sin(\boldsymbol{x}^u).$$

□

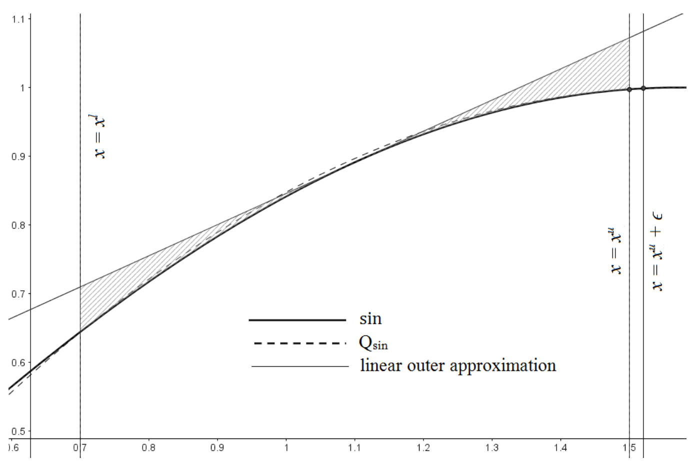


Figure 5.4: For the sine function, we compare a linear outer approximation to the new quadratic relaxation defined by the points $(\boldsymbol{x}^l; \sin(\boldsymbol{x}^l))$, $(\boldsymbol{x}^u; \sin(\boldsymbol{x}^u))$, and $(\boldsymbol{x}^u + \epsilon; \sin(\boldsymbol{x}^u + \epsilon))$

Proposition 5.2. *Given* ϵ *s.t.* $0 < \epsilon < \frac{\pi}{2} - \boldsymbol{x}^u$*, if* $0 \leqslant \boldsymbol{x}^l \leqslant \boldsymbol{x}^u < \frac{\pi}{2}$*, then*

$$\sin(x) \leqslant Q_{\sin}(\boldsymbol{x}^l, \boldsymbol{x}^u, \boldsymbol{x}^u + \epsilon), \ \forall x \in [\boldsymbol{x}^l, \boldsymbol{x}^u].$$

Proof. We have $\boldsymbol{x}_1 = \boldsymbol{x}^l$, $\boldsymbol{x}_2 = \boldsymbol{x}^u$ and $\boldsymbol{x}_3 = \boldsymbol{x}^u + \epsilon$. This leads to $\boldsymbol{\delta}_{32} = \boldsymbol{x}_u + \epsilon - \boldsymbol{x}_u = \epsilon$ and $\boldsymbol{\delta}_{31} = (\boldsymbol{x}_u + \epsilon) - \boldsymbol{x}_l = \boldsymbol{\delta}_{21} + \epsilon$. Consider the function corresponding to the difference between $Q_{\sin}$ and sin,

$$f_\epsilon(x) = Q_{\sin}(\boldsymbol{x}^l, \boldsymbol{x}^u, \boldsymbol{x}^u + \epsilon) - \sin(x)$$

$$= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\boldsymbol{\delta}_{21}^2\epsilon + \boldsymbol{\delta}_{21}\epsilon^2}(x - \boldsymbol{x}_1)(x - \boldsymbol{x}_2) + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}}(x - \boldsymbol{x}_2) + \sin(\boldsymbol{x}_2) - \sin(x).$$

We will first show that $f_\epsilon(x)$ is strictly decreasing at $\boldsymbol{x}^u$. Since $f_\epsilon(\boldsymbol{x}^u) = 0$, this implies that f is positive in the left neighbourhood of $\boldsymbol{x}^u$. We will then prove that $f_\epsilon(x)$ has a unique stationary point in the interval $[\boldsymbol{x}^l, \boldsymbol{x}^u]$. Since $f_\epsilon(\boldsymbol{x}^l) = f_\epsilon(\boldsymbol{x}^u) = 0$, this is sufficient to prove that $f_\epsilon(x)$ is positive on the whole interval.

Let us consider the derivative of $f_\epsilon(x)$,

$$f_\epsilon^{'}(x) = \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\boldsymbol{\delta}_{21}^2\epsilon + \boldsymbol{\delta}_{21}\epsilon^2}(2x - \boldsymbol{x}_1 - \boldsymbol{x}_2) + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(x).$$

Now consider $f_\epsilon^{'}(\boldsymbol{x}^u) = f_\epsilon^{'}(\boldsymbol{x}_2)$,

$$\begin{aligned}
f_\epsilon^{'}(\boldsymbol{x}_2) &= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\boldsymbol{\delta}_{21}^2\epsilon + \boldsymbol{\delta}_{21}\epsilon^2}(2\boldsymbol{x}_2 - \boldsymbol{x}_1 - \boldsymbol{x}_2) + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2) \\
&= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\boldsymbol{\delta}_{21}^2\epsilon + \boldsymbol{\delta}_{21}\epsilon^2}(\boldsymbol{x}_2 - \boldsymbol{x}_1) + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2) \\
&= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\boldsymbol{\delta}_{21}^2\epsilon + \boldsymbol{\delta}_{21}\epsilon^2}\boldsymbol{\delta}_{21} + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2) \\
&= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \boldsymbol{\phi}_{21}\epsilon}{\epsilon(\boldsymbol{\delta}_{21} + \epsilon)} + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2) \\
&= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21}}{\epsilon(\boldsymbol{\delta}_{21} + \epsilon)} - \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21} + \epsilon} + \frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2) \\
&= \frac{\boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \epsilon\boldsymbol{\phi}_{21} + \epsilon(\boldsymbol{\delta}_{21} + \epsilon)\left(\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2)\right)}{\epsilon(\boldsymbol{\delta}_{21} + \epsilon)} = \frac{h(\epsilon)}{\epsilon(\boldsymbol{\delta}_{21} + \epsilon)},
\end{aligned}$$

where

$$h(\epsilon) = \boldsymbol{\phi}_{32}\boldsymbol{\delta}_{21} - \epsilon\boldsymbol{\phi}_{21} + \epsilon(\boldsymbol{\delta}_{21} + \epsilon)\left(\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2)\right).$$

Since $\epsilon(\boldsymbol{\delta}_{21} + \epsilon) > 0$, we have that $f_\epsilon^{'}(\boldsymbol{x}_2) \leqslant 0 \Leftrightarrow h'(\epsilon) \leqslant 0$.

Consider the derivative of h,

$$\begin{aligned}
h'(\epsilon) &= \boldsymbol{\delta}_{21}\cos(\boldsymbol{x}_2 + \epsilon) - \boldsymbol{\phi}_{21} + (\boldsymbol{\delta}_{21} + 2\epsilon)\left(\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2)\right) \\
&= \boldsymbol{\delta}_{21}\left(\cos(\boldsymbol{x}_2 + \epsilon) - \cos(\boldsymbol{x}_2)\right) + 2\epsilon\left(\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2)\right).
\end{aligned}$$

Based on Proposition 5.1, we have that $\cos(\boldsymbol{x}_2 + \epsilon) - \cos(\boldsymbol{x}_2) \leqslant -\epsilon\sin(\boldsymbol{x}_2)$, consequently,

$$\begin{aligned}
h'(\epsilon) &\leqslant -\epsilon\boldsymbol{\delta}_{21}\sin(\boldsymbol{x}_2) + 2\epsilon\left(\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - \cos(\boldsymbol{x}_2)\right) \\
&= \epsilon\left(2\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - 2\cos(\boldsymbol{x}_2) - \boldsymbol{\delta}_{21}\sin(\boldsymbol{x}_2)\right).
\end{aligned}$$

We will next show that
$$2\frac{\boldsymbol{\phi}_{21}}{\boldsymbol{\delta}_{21}} - 2\cos(\boldsymbol{x}_2) - \boldsymbol{\delta}_{21}\sin(\boldsymbol{x}_2) \leqslant 0,$$
or, equivalently,
$$\begin{aligned} g(\boldsymbol{\delta}_{21}) &= \boldsymbol{\phi}_{21} - \boldsymbol{\delta}_{21}\cos(\boldsymbol{x}_2) - \frac{1}{2}\boldsymbol{\delta}_{21}^2\sin(\boldsymbol{x}_2) \\ &= \sin(\boldsymbol{x}_2) - \sin(\boldsymbol{x}_2 - \boldsymbol{\delta}_{21}) - \boldsymbol{\delta}_{21}\cos(\boldsymbol{x}_2) - \frac{1}{2}\boldsymbol{\delta}_{21}^2\sin(\boldsymbol{x}_2) \leqslant 0. \end{aligned}$$
Consider the derivatives:
$$g'(\boldsymbol{\delta}_{21}) = \cos(\boldsymbol{x}_2 - \boldsymbol{\delta}_{21}) - \cos(\boldsymbol{x}_2) - \boldsymbol{\delta}_{21}\sin(\boldsymbol{x}_2),$$
$$g''(\boldsymbol{\delta}_{21}) = \sin(\boldsymbol{x}_2 - \boldsymbol{\delta}_{21}) - \sin(\boldsymbol{x}_2) < 0.$$
Since $g(0) = 0$, $g'(0) = 0$ and $g''(\boldsymbol{\delta}_{21}) < 0$, we have proved that $g(\boldsymbol{\delta}_{21}) \leqslant 0,\ \forall \boldsymbol{\delta}_{21} \geqslant 0$ and thus $f_\epsilon^{'}(\boldsymbol{x}_2) \leqslant 0,\ \forall \epsilon,\ 0 < \epsilon < \frac{\pi}{2} - \boldsymbol{x}^u$. Since $f_\epsilon^{'}(x)$ is a convex function and is negative at the upper bound $\boldsymbol{x}^u$, it can have at most one root in the interval $[\boldsymbol{x}^l, \boldsymbol{x}^u]$. Consequently f_ϵ has a unique stationary point in this interval. Since $f_\epsilon(\boldsymbol{x}^l) = f_\epsilon(\boldsymbol{x}^u) = 0$, and f_ϵ is positive in the left neighbourhood of $\boldsymbol{x}^u$, it is positive on the whole interval.

□

Note that this proof can be easily adapted to the case $f(x) = \cos(x),\ x \in [-\frac{\pi}{2}, 0]$ by translating the x axis by $\frac{\pi}{2}$. It can also be adapted to $\cos(x),\ x \in [0, \frac{\pi}{2}]$ and $\sin(x),\ x \in [-\frac{\pi}{2}, 0]$ by inverting the sign of x.

Having a quadratic relaxation for $\sin(x)$ and $\cos(x)$ enables us to use the convex hull formulation of quadratic on/off constraints introduced in Section 5.4.

5.6 Application: Optimal Transmission Switching

In this section, our new results along with some additional improvements are used to strengthen the Convex Quadratic relaxation of the Optimal Transmission Switching problem that was discussed in Subsection 2.6.1 of Chapter 2.

Let us restate the formulation of the QC-OTS problem:

Model 5.1: The Quadratic Convex relaxation of the Optimal Transmission Switching problem

---

variables for each $(i, j) \in E$:

$p_{ij},\ q_{ij}$

$cs_{ij} \in [\cos(\boldsymbol{\theta}_{ij}^u),\ 1;\ 0,\ 0],\ sn_{ij} \in [-\sin(\boldsymbol{\theta}_{ij}^u),\ \sin(\boldsymbol{\theta}_{ij}^u);\ 0,\ 0]$

$w_{ij} \in [\boldsymbol{v}_i^l\boldsymbol{v}_j^l,\ \boldsymbol{v}_i^u\boldsymbol{v}_j^u;\ 0,\ 0],\ w_{ij}^R \in [\boldsymbol{v}_i^l\boldsymbol{v}_j^l\cos(\boldsymbol{\theta}_{ij}^u),\ \boldsymbol{v}_i^u\boldsymbol{v}_j^u;\ 0,\ 0]$

$w_{ij}^I \in [-\boldsymbol{v}_i^l\boldsymbol{v}_j^l\sin(\boldsymbol{\theta}_{ij}^u),\ \boldsymbol{v}_i^u\boldsymbol{v}_j^u\sin(\boldsymbol{\theta}^u);\ 0,\ 0]$

$\theta_{ij} \in [-\boldsymbol{\theta}^u_{ij},\ \boldsymbol{\theta}^u_{ij};\ \ -\boldsymbol{M}^u,\ \boldsymbol{M}^u]$

$z_{ij} \in \{0,1\}$

variables for each $\boldsymbol{i \in N}$:

$w_i \in [(\boldsymbol{v}^l_i)^2,\ (\boldsymbol{v}^u_i)^2]$

$p^g_i \in [\boldsymbol{p}^{\boldsymbol{gl}}_i, \boldsymbol{p}^{\boldsymbol{gu}}_i],\ q^g_i \in [\boldsymbol{q}^{\boldsymbol{gl}}_i,\ \boldsymbol{q}^{\boldsymbol{gu}}_i]$

objective:

$$\min \sum_{i\ \in G} \left(\boldsymbol{c}_{2i}(p^g_i)^2 + \boldsymbol{c}_{1i}(p^g_i) + \boldsymbol{c}_{0i}\right)$$

subject to

$$\theta_r = 0 \tag{5.7a}$$

$$p^g_i - \boldsymbol{p}^d_i = \sum_{(i,j)\in E} p_{ij} + \sum_{(j,i)\in E} p_{ij} \qquad \forall i \in N \tag{5.7b}$$

$$q^g_i - \boldsymbol{q}^d_i = \sum_{(i,j)\in E} q_{ij} + \sum_{(j,i)\in E} q_{ij} \qquad \forall i \in N \tag{5.7c}$$

$$p_{ij} = \mathbf{g}_{ij} w_i - \mathbf{g}_{ij} w^R_{ij} - \mathbf{b}_{ij} w^I_{ij} \qquad \forall (i,j) \in E \tag{5.7d}$$

$$q_{ij} = -\mathbf{b}_{ij} w_i + \mathbf{b}_{ij} w^R_{ij} - \mathbf{g}_{ij} w^I_{ij} \qquad \forall (i,j) \in E \tag{5.7e}$$

$$-\tan(\boldsymbol{\theta}^{\boldsymbol{u}}_{ij}) w^R_{ij} \leqslant w^I_{ij} \leqslant \tan(\boldsymbol{\theta}^{\boldsymbol{u}}_{ij}) w^R_{ij} \qquad \forall (i,j) \in E \tag{5.7f}$$

$$(cs_{ij}, \theta_{ij}, z_{ij}) \in CS^{0\text{-}1}_{ij} \qquad \forall (i,j) \in E \tag{5.7g}$$

$$(sn_{ij}, \theta_{ij}, z_{ij}) \in SN^{0\text{-}1}_{ij} \qquad \forall (i,j) \in E \tag{5.7h}$$

$$v^2_i \leqslant w_i \leqslant (\boldsymbol{v}^l_i + \boldsymbol{v}^u_i) v_i - \boldsymbol{v}^l_i \boldsymbol{v}^u_i \qquad \forall i \in N \tag{5.7i}$$

$$w_{ij} \in MC(v_i, v_j) \ \text{ if } z_{ij} = 1 \qquad \forall (i,j) \in E \tag{5.7j}$$

$$w^R_{ij} \in MC(Q^{\cos}_{ij}, w_{ij}) \ \text{ if } z_{ij} = 1 \qquad \forall (i,j) \in E \tag{5.7k}$$

$$w^I_{ij} \in MC(Q^{\sin}_{ij}, w_{ij}) \ \text{ if } z_{ij} = 1 \qquad \forall (i,j) \in E \tag{5.7l}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{s}^u_{ij} z_{ij} \qquad \forall (i,j) \in E \tag{5.7m}$$

$$p^2_{ij} + q^2_{ij} \leqslant (\boldsymbol{v}^u)^2 l_{ij} z_{ij} \qquad \forall (i,j) \in E \tag{5.7n}$$

$$p^2_{ij} + q^2_{ij} \leqslant \boldsymbol{l}^u_{ij} w_i z_{ij} \qquad \forall (i,j) \in E \tag{5.7o}$$

$$l_{ij} = (\boldsymbol{g}^2_{ij} + \boldsymbol{b}^2_{ij})(w_i + w_j - 2w^R_{ij}) \qquad \forall (i,j) \in E \tag{5.7p}$$

In this model, $x_{ij} \in [\boldsymbol{x}^{l1}_{ij}, \boldsymbol{x}^{u1}_{ij}; \boldsymbol{x}^{l0}_{ij}, \boldsymbol{x}^{u0}_{ij}]$ defines a variable with on/off bounds and $\widehat{cs}(\theta_i - \theta_j)$, $\widetilde{sn}(\theta_i - \theta_j)$ and $\widehat{sn}(\theta_i - \theta_j)$ represent the under- and overestimating functions used to build convex relaxations of trigonometric functions (see Subsection 2.6.1).

For the description of the nonconvex AC-OTS problem we refer the reader to Section 2.5.

5.6.1 Asymmetrical bounds

When the bounds on the phase angle difference $(\boldsymbol{\theta}^l_{ij}, \boldsymbol{\theta}^u_{ij})$ are asymmetrical, the expressions for the bounds on the auxiliary variables become different from those given in Model 5.1.

For the variables cs_{ij} representing the relaxation of the cosine function the updated bounds are given by:

$$\boldsymbol{cs}_{ij}^{l} = \min(\cos(\boldsymbol{\theta}_{ij}^{l}), \cos(\boldsymbol{\theta}_{ij}^{u})),$$

$$\boldsymbol{cs}_{ij}^{u} = \begin{cases} 1 \text{ if } \boldsymbol{\theta}_{ij}^{l} \leqslant 0 \text{ and } \boldsymbol{\theta}_{ij}^{u} \geqslant 0, \\ \max(\cos(\boldsymbol{\theta}_{ij}^{l}), \cos(\boldsymbol{\theta}_{ij}^{u})) \text{ otherwise.} \end{cases}$$

The bounds on the variables w_{ij}^{R} that capture the trilinear products $v_i v_j \cos(\theta_{ij})$ can be easily expressed through the above. For the bounds on w_{ij}^{I} we have:

$$(\boldsymbol{w}_{ij}^{I})^{l} = \begin{cases} \boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}_{ij}^{l}) \text{ if } \boldsymbol{\theta}_{ij}^{l} \leqslant 0, \\ \boldsymbol{v}_i^l \boldsymbol{v}_j^l \sin(\boldsymbol{\theta}_{ij}^{l}) \text{ if } \boldsymbol{\theta}_{ij}^{l} > 0, \end{cases}$$

$$(\boldsymbol{w}_{ij}^{I})^{u} = \begin{cases} \boldsymbol{v}_i^u \boldsymbol{v}_j^u \sin(\boldsymbol{\theta}_{ij}^{u}) \text{ if } \boldsymbol{\theta}_{ij}^{u} \geqslant 0, \\ \boldsymbol{v}_i^l \boldsymbol{v}_j^l \sin(\boldsymbol{\theta}_{ij}^{u}) \text{ if } \boldsymbol{\theta}_{ij}^{u} < 0. \end{cases}$$

This generalisation ensures that after applying bound tightening that will be discussed in Subsection 5.6.6 below, the bounds on these variables remain valid.

5.6.2 Tightening the big-M constants

The following proposition provides expressions for improved big-M constants that will result in tighter continuous relaxations.

Proposition 5.3. *Let E^u (resp. E^l) denote the set of $|N|-1$ edges having the largest upper (resp. smallest lower) bound on the phase angle difference θ_{ij}. Then,*

$$\theta_i - \theta_j \leqslant \sum_{E^u} \boldsymbol{\theta}_{ij}^{u}, \text{ and } \theta_i - \theta_j \geqslant \sum_{E^l} \boldsymbol{\theta}_{ij}^{l}, \ \forall (i,j) \in E.$$

Proof. Due to Kirchhoff's Voltage Law, the voltage drop around a loop is zero. Observe that the longest loop-less path has at most $|N|-1$ edges. Hence the voltage drop $\theta_i - \theta_j$ cannot be larger than the sum of the largest $(|N|-1)$ values of $\boldsymbol{\theta}_{ij}^{u}$. A similar argument holds for the lower bound. □

5.6.3 Relaxations of trigonometric on/off constraints

The quadratic relaxation [100] for $\cos(\theta_{ij})$ does not support asymmetrical phase angle bounds. Furthermore, the on/off version of these quadratic constraints are formulated using the big-M approach. In light of the results presented in previous sections, we are able to improve the QC relaxation using asymmetrical quadratic relaxations and tight on/off constraints representation. The complete relaxations of the trigonometric terms can be found in Appendix A. As a showcase, we present below the formulation of the on/off version corresponding to the quadratic relaxation of $\sin(\theta_{ij})$ when $\boldsymbol{\theta}_{ij}^{u} \leqslant 0$. Similar constraints can be generated for the other cases. Let sn_{ij} denote the auxiliary variable used in the quadratic relaxation corresponding to $\sin(\theta_{ij})$, we have,

$$\left\{\begin{array}{l} sn_{ij} \geqslant q^{\sin}(\theta_{ij}), \\ \sin(\boldsymbol{\theta}^l_{ij}) \leqslant sn_{ij} \leqslant \sin(\boldsymbol{\theta}^u_{ij}), \\ \boldsymbol{\theta}^l_{ij} \leqslant \theta_{ij} \leqslant \boldsymbol{\theta}^u_{ij}, \\ z_{ij} = 1 \end{array}\right\} \vee \left\{\begin{array}{l} sn_{ij} = 0, \\ \sum_{E^l} \boldsymbol{\theta}^l_{ij} \leqslant \theta_{ij} \leqslant \sum_{E^u} \boldsymbol{\theta}^u_{ij}, \\ z_{ij} = 0 \end{array}\right\},$$

where $q^{\sin}(\theta_{ij}) = \boldsymbol{a}\theta^2_{ij} + \boldsymbol{b}\theta_{ij} + \boldsymbol{c}$ is the quadratic approximation of the sine function introduced in Section 5.5.

Based on Theorem 5.2, we can write the convex hull formulation of this disjunction as follows,

$$\begin{aligned} &\theta_{ij} - \sum_{E^u} \boldsymbol{\theta}^u_{ij}\,(1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{sn_{ij}z + \delta z^2}{\boldsymbol{a}}}, \\ &\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c}. \end{aligned} \tag{5.8}$$

Note that this formulation is nondifferentiable at points where $sn_{ij}z_{ij} + \boldsymbol{\delta} z^2_{ij} = 0$. Numerical issues arising from this irregularity can be alleviated using a linear outer approximation of the nonlinear constraints. The outer approximation is obtained by finding tangent planes to the boundary surfaces of constraints (5.8) at points $\theta_{ij} \in [\boldsymbol{\theta}^l_{ij}, \boldsymbol{\theta}^u_{ij}]$ and $z_{ij} \in [0, 1]$. This results in a relaxation which is still valid as the functions are convex.

In particular, given a point $(\bar{\boldsymbol{\theta}}_{ij}, \bar{\boldsymbol{z}}_{ij}, \bar{\boldsymbol{sn}}_{ij})$ on the surface of (5.8), the corresponding linearised constraint is given by:

$$\nabla h^T_c(\bar{\boldsymbol{\theta}}_{ij}, \bar{\boldsymbol{z}}_{ij}, \bar{\boldsymbol{sn}}_{ij}) \begin{pmatrix} \theta_{ij} - \bar{\boldsymbol{\theta}}_{ij} \\ z_{ij} - \bar{\boldsymbol{z}}_{ij} \\ sn_{ij} - \bar{\boldsymbol{sn}}_{ij} \end{pmatrix} + h_c(\bar{\boldsymbol{\theta}}_{ij}, \bar{\boldsymbol{z}}_{ij}, \bar{\boldsymbol{sn}}_{ij}) \leqslant 0,$$

where

$$h_c(\theta_{ij}, z_{ij}, sn_{ij}) = \theta_{ij} - \sum_{E^u} \boldsymbol{\theta}^u_{ij}\,(1-z) + \boldsymbol{\rho} z - \sqrt{\frac{sn_{ij}z + \delta z^2}{\boldsymbol{a}}}.$$

Let us emphasise that these constraints can be used with solvers for quadratically constrained models such as, for example, Gurobi.

The feasible sets of the on/off relaxations constructed as described above will be denoted by $QSN^{0\text{-}1}_{ij}$ and $QCS^{0\text{-}1}_{ij}$.

5.6.4 On/off lifted nonlinear cuts

In this subsection, we recall the result that was independently presented by Coffrin et al. [42] and Chen et al. [35]. Consider the nonconvex constraint:

$$(w^R_{ij})^2 + (w^I_{ij})^2 = w_i w_j.$$

Observe that this is a valid equality in AC-OPF. The well-known second-order cone relaxation $w_{ij}^R + w_{ij}^I \leqslant w_i w_j$ yields a tight upper bound on $w_{ij}^R + w_{ij}^I$. The cuts presented in this subsection are providing a lower bound.

Consider the nonconvex voltage feasibility set:

$$(\boldsymbol{v}_i^l)^2 \leqslant w_i \leqslant (\boldsymbol{v}_i^u)^2, \tag{5.9a}$$
$$(\boldsymbol{v}_j^l)^2 \leqslant w_j \leqslant (\boldsymbol{v}_j^u)^2, \tag{5.9b}$$
$$\boldsymbol{w}_{ij}^{Rl} \leqslant w_{ij}^R \leqslant \boldsymbol{w}_{ij}^{Ru}, \tag{5.9c}$$
$$\boldsymbol{w}_{ij}^{Il} \leqslant w_{ij}^I \leqslant \boldsymbol{w}_{ij}^{Iu}, \tag{5.9d}$$
$$\tan(\boldsymbol{\theta}_{ij}^l) w_{ij}^R \leqslant w_{ij}^I \leqslant \tan(\boldsymbol{\theta}_{ij}^u) w_{ij}^R, \tag{5.9e}$$
$$(w_{ij}^R)^2 + (w_{ij}^I)^2 = w_i w_j. \tag{5.9f}$$

Note that this set can be reformulated in three dimensions by using equation (5.9f) to eliminate the w_j variable. Then (5.9b, 5.9f) can be substituted by:

$$w_i (\boldsymbol{v}_j^l)^2 \leqslant (w_{ij}^R)^2 + (w_{ij}^I)^2 \leqslant w_i (\boldsymbol{v}_j^u)^2. \tag{5.10}$$

We will use an alternate representation of the voltage angle bounds. Specifically, given $-\pi/2 \leqslant \boldsymbol{\theta}_{ij}^l < \boldsymbol{\theta}_{ij}^u \leqslant \pi/2$, we define the following constants:

$$\boldsymbol{\phi}_{ij} = (\boldsymbol{\theta}_{ij}^u + \boldsymbol{\theta}_{ij}^l)/2, \tag{5.11a}$$
$$\boldsymbol{\delta}_{ij} = (\boldsymbol{\theta}_{ij}^u - \boldsymbol{\theta}_{ij}^l)/2, \tag{5.11b}$$
$$\boldsymbol{v}_i^{\boldsymbol{\sigma}} = \boldsymbol{v}_i^{\boldsymbol{l}} + \boldsymbol{v}_i^{\boldsymbol{u}}, \tag{5.11c}$$
$$\boldsymbol{v}_j^{\boldsymbol{\sigma}} = \boldsymbol{v}_j^{\boldsymbol{l}} + \boldsymbol{v}_j^{\boldsymbol{u}}. \tag{5.11d}$$

First, a convex relaxation of the set described by (5.9) is obtained by replacing the lower bound in (5.10) by a linear inequality:

$$\boldsymbol{v}_j^l \cos(\boldsymbol{\delta}_{ij}) w_i - \boldsymbol{v}_i^\sigma \cos(\boldsymbol{\phi}_{ij}) w_{ij}^R - \boldsymbol{v}_i^\sigma \sin(\boldsymbol{\phi}_{ij}) w_{ij}^I + \boldsymbol{v}_i^l \boldsymbol{v}_i^u \boldsymbol{v}_j^l \cos(\boldsymbol{\delta}_{ij}) \leqslant 0. \tag{5.12}$$

It can be proven [42] that cut (5.12) is valid for set (5.9). Let S_c denote the resulting relaxation:

$$S_c = \left\{ (w_{ij}^R, w_{ij}^I, w_i) \in \mathbb{R}^3 \mid \begin{array}{c} (5.9a), (5.9c) - (5.9e), (5.12) \\ (w_{ij}^R)^2 + (w_{ij}^I)^2 \leqslant w_i (\boldsymbol{v}_j^u)^2 \end{array} \right\}$$

Set S_c is utilised to develop two more nonlinear cuts:

$$\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i -$$

$$\boldsymbol{v}_i^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} \frac{(w_{ij}^R)^2 + (w_{ij}^I)^2}{w_i} \geqslant \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij})(\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} - \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}}) \tag{5.13a}$$

$$\begin{aligned} &\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i - \\ &\boldsymbol{v}_i^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} \frac{(w_{ij}^R)^2 + (w_{ij}^I)^2}{w_i} \geqslant -\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij})(\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} - \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}}) \end{aligned} \tag{5.13b}$$

In the three dimensional space these cuts are redundant with respect to cut (5.12). However, as equation (5.9f) is relaxed into an inequality, the problem is lifted into a four dimensional space $(w_{ij}^R, w_{ij}^I, w_i, w_j)$:

$$\begin{aligned} &\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i - \\ &\boldsymbol{v}_i^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} w_j \geqslant \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij})(\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} - \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}}) \quad \forall (i,j) \in E \end{aligned} \tag{5.14a}$$

$$\begin{aligned} &\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i - \\ &\boldsymbol{v}_i^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} w_j \geqslant -\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij})(\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} - \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}}) \quad \forall (i,j) \in E \end{aligned} \tag{5.14b}$$

Cuts (5.14a), (5.14b) dominate cut (5.12) in the four dimensional space [42].

Cuts (5.14a), (5.14b) were derived for the continuous model. To adapt them for the mixed-integer case, it is necessary to make sure that these constraints are valid when the corresponding line is deactivated. This is achieved by enforcing the cuts only when the line is switched on. To express the resulting disjunctive constraint algebraically, we apply formulation (5.6).

5.6.5 The complete strengthened QC-OTS model

Now when all the improvements have been introduced, the complete strengthened QC-OTS model can be written as:

Model 5.2: Full strengthened QC-OTS

---

variables for each $(i, j) \in E$:

$p_{ij},\ q_{ij}$

$cs_{ij} \in [\boldsymbol{cs}_{ij}^l,\ \boldsymbol{cs}_{ij}^u;\ 0,\ 0],\ sn_{ij} \in [\sin(\boldsymbol{\theta}_{ij}^l),\ \sin(\boldsymbol{\theta}_{ij}^u);\ 0,\ 0]$

$w_{ij} \in [\boldsymbol{v}_i^l \boldsymbol{v}_j^l,\ \boldsymbol{v}_i^u \boldsymbol{v}_j^u;\ 0,\ 0],\ w_{ij}^R \in [\boldsymbol{v}_i^l \boldsymbol{v}_j^l \boldsymbol{cs}_{ij}^l,\ \boldsymbol{v}_i^u \boldsymbol{v}_j^u \boldsymbol{cs}_{ij}^u;\ 0,\ 0]$

$w_{ij}^I \in [(\boldsymbol{w}_{ij}^I)^l,\ (\boldsymbol{w}_{ij}^I)^u;\ 0,\ 0]$

$\theta_{ij} \in [-\boldsymbol{\theta}_{ij}^u,\ \boldsymbol{\theta}_{ij}^u;\ \boldsymbol{M}^l,\ \boldsymbol{M}^u]$

$z_{ij} \in \{0, 1\}$

variables for each $i \in N$:

$w_i \in [(\boldsymbol{v}_i^l)^2,\ (\boldsymbol{v}_i^u)^2]$

$p_i^g \in [\boldsymbol{p}_i^{\boldsymbol{gl}}, \boldsymbol{p}_i^{\boldsymbol{gu}}],\ q_i^g \in [\boldsymbol{q}_i^{\boldsymbol{gl}},\ \boldsymbol{q}_i^{\boldsymbol{gu}}]$

objective:

$$\min \sum_{i\ \in G} \left(\boldsymbol{c}_{2i}(p_i^g)^2 + \boldsymbol{c}_{1i}(p_i^g) + \boldsymbol{c}_{0i} \right)$$

subject to (5.15a)

$$p_i^g - \mathbf{p}_i^l = \sum_{(i,j)\in E} p_{ij}\ \ \forall i \in N \tag{5.15b}$$

$$q_i^g - \mathbf{q}_i^l = \sum_{(i,j)\in E} q_{ij}\ \ \forall i \in N \tag{5.15c}$$

$$p_{ij} = \boldsymbol{g}_{ij} w_i - \boldsymbol{g}_{ij} w_{ij}^R - \boldsymbol{b}_{ij} w_{ij}^I\ \ \forall (i,j) \in E \tag{5.15d}$$

$$q_{ij} = -\boldsymbol{b}_{ij} w_i + \boldsymbol{b}_{ij} w_{ij}^R - \boldsymbol{g}_{ij} w_{ij}^I\ \ \forall (i,j) \in E \tag{5.15e}$$

$$v_i^2 \leqslant w_i \leqslant (\boldsymbol{v}_i^l + \boldsymbol{v}_i^u) v_i - \boldsymbol{v}_i^l \boldsymbol{v}_i^u\ \ \forall i \in N \tag{5.15f}$$

$$(cs_{ij}, \theta_{ij}, z_{ij}) \in QCS_{ij}^{0\text{-}1}\ \ \forall (i,j) \in E \tag{5.15g}$$

$$(sn_{ij}, \theta_{ij}, z_{ij}) \in QSN_{ij}^{0\text{-}1}\ \ \forall (i,j) \in E \tag{5.15h}$$

$$w_{ij} \in MC(v_i, v_j)\ \ \text{if } z_{ij} = 1\ \forall (i,j) \in E \tag{5.15i}$$

$$w_{ij}^R \in MC(cs_{ij}, w_{ij})\ \ \text{if } z_{ij} = 1\ \forall (i,j) \in E \tag{5.15j}$$

$$w_{ij}^I \in MC(sn_{ij}, w_{ij})\ \ \text{if } z_{ij} = 1\ \forall (i,j) \in E \tag{5.15k}$$

$$p_{ij}^2 + q_{ij}^2 \leqslant \boldsymbol{s}_{ij}^u z_{ij}\ \ \forall (i,j) \in E \tag{5.15l}$$

$$p_{ij}^2 + q_{ij}^2 \leqslant (\boldsymbol{v}^u)^2 l_{ij} z_{ij}\ \ \forall (i,j) \in E \tag{5.15m}$$

$$p_{ij}^2 + q_{ij}^2 \leqslant \boldsymbol{l}_{ij}^u w_i z_{ij}\ \ \forall (i,j) \in E \tag{5.15n}$$

$$l_{ij} = (\boldsymbol{g}_{ij}^2 + \boldsymbol{b}_{ij}^2)(w_i + w_j - 2w_{ij}^R)\ \ \forall (i,j) \in E \tag{5.15o}$$

$$\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i -$$
$$\boldsymbol{v}_i^{\boldsymbol{u}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} w_j \geqslant \boldsymbol{v}_i^u \boldsymbol{v}_j^u \boldsymbol{c}^{lnc} \quad \text{if } z_{ij} = 1\ \forall (i,j) \in E \tag{5.15p}$$

$$\boldsymbol{v}_i^{\boldsymbol{\sigma}} \boldsymbol{v}_j^{\boldsymbol{\sigma}} (w_{ij}^R \cos(\boldsymbol{\phi}_{ij}) + w_{ij}^I \sin(\boldsymbol{\phi}_{ij})) - \boldsymbol{v}_j^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_j^{\boldsymbol{\sigma}} w_i -$$
$$\boldsymbol{v}_i^{\boldsymbol{l}} \cos(\boldsymbol{\delta}_{ij}) \boldsymbol{v}_i^{\boldsymbol{\sigma}} w_j \geqslant -\boldsymbol{v}_i^l \boldsymbol{v}_j^l \boldsymbol{c}^{lnc} \quad \text{if } z_{ij} = 1\ \forall (i,j) \in E, \tag{5.15q}$$

---

where $\boldsymbol{cs}_{ij}^l$, $\boldsymbol{cs}_{ij}^u$, $(\boldsymbol{w}_{ij}^I)^l$ and $(\boldsymbol{w}_{ij}^I)^u$ are the bounds defined in Subsection 5.6.1, the constant $\boldsymbol{c}^{lnc}$ is given by

$$\boldsymbol{c}^{lnc} = \cos(\boldsymbol{\delta}_{ij})(\boldsymbol{v}_i^{\boldsymbol{l}} \boldsymbol{v}_j^{\boldsymbol{l}} - \boldsymbol{v}_i^{\boldsymbol{u}} \boldsymbol{v}_j^{\boldsymbol{u}}),$$

$\boldsymbol{M}^l$ and $\boldsymbol{M}^u$ are big-M constants introduced in Subsection 5.6.2, MC denotes the McCormick relaxation [136] of a product of two variables, sn_{ij} and cs_{ij} denote quadratic relaxations of the sine and cosine functions respectively. (5.15l) is the on/off version of the thermal limit constraint and (5.15m)-(5.15o) are on/off current magnitude constraints that are used to further strengthen the model. Disjunctive versions of linear constraints are obtained by applying formulation (5.6).

For the full description of the sine and cosine relaxations, see Appendix A. The implementation can be found in the PowerTools repository at `https://github.com/hhijazi/PowerTools`.

5.6.6 Bound propagation

The strength of the QC relaxation depends on the bounds on voltage magnitudes and phase angle differences. In order to exploit this feature we apply bound propagation to the QC-OTS model, as was first proposed by Coffrin et al. [43] for the QC relaxation of the continuous Optimal Power Flow model.

For this purpose the traditional constraint-programming notions, such as minimal continuous constraint networks (CCNs) and bound-consistency, are adapted to relaxations by defining the concept of a continuous constraint relaxation network (CCRN) [43]. Algorithms for computing minimal and bound-consistent CCRNs are introduced.

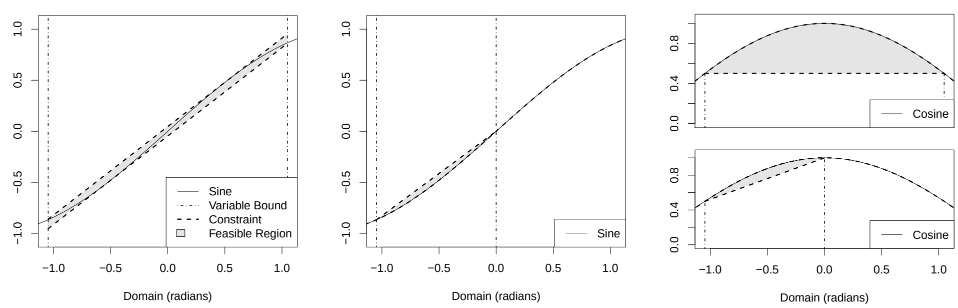


Figure 5.5: The impact of variable bounds on the convex relaxations

In this chapter we use minimal CCRNs, because they yield tighter bounds than bound-consistent networks. In paper [43], the minCCRN algorithm was used to propagate the bounds on θ_{ij} and v_i in the continuous QC model. To avoid solving many mixed-integer programs, in the revised minCCRN algorithm we find solutions of the continuous relaxations of the original programs. In addition to θ_{ij} and v_i, bound propagation on the binary variables is introduced: if the lower bound of z_{ij} in the relaxed model is proven to be greater than 0, then this variable can be fixed to 1.

Finding a minimal CCRN involves solving problems of minimizing the upper bound and maximizing the lower bound for each variable θ_{ij} and v_i and maximizing the lower bound for each z_{ij}. This is repeated until a fixpoint is reached. An important observation is that since the problems associated with each variable are independent from each other, the algorithm can be fully parallelised with one thread corresponding to each variable.

5.7 Computational Results

This section evaluates the proposed algorithms on 23 test cases from the Power Grid Library - v17.08 (https://github.com/power-grid-lib/pglib-opf) ranging from 3 to 300 nodes. The models were implemented in C++ and solved using Gurobi 7.5.0 on Dell PowerEdge M630 machines with 64GB of memory and Intel Xeon E5-2660 V3 processors running at 2.6 GHz with 25 MB cache using 5 cores per process.

For these experiments instances with optimality gaps $\geqslant 1\%$ yielded by the previous state-of-the-art QC-OTS model were chosen. The gaps are calculated as the relative differences between the upper bounds obtained from solving the exact AC-OTS model and the lower bounds from the QC relaxations of the OTS model. The upper bound for each instance is the minimum between the solutions of two heuristics. One computes a local optimum of the nonconvex AC-OTS model using Bonmin-1.8.4. The other evaluates the upper bound by solving the AC-OTS model in the Ipopt 3.12 [187] solver after fixing the binary variables' values to those obtained from solving the QC-OTS relaxation. Each algorithm was run with a time limit of 7200s.

In all tables in this Section, "ERR" means that the solver reported numerical issues and the model was not solved and the '-' symbol means that the upper and/or lower bound was not computed due to the solver reaching the time limit.

5.7.1 Bound propagation strength and performance

Table 5.1: Bound propagation results

Test Case	Time (s)	Parallel time(s)	θ domain (%)	v domain (%)	Free lines (%)
pglib_opf_case3_lmbd	0.14	0.11	41.9	100.0	33.33
pglib_opf_case5_pjm	0.77	0.26	16.8	99.1	100
pglib_opf_case30_ieee	21.61	1.74	16.6	94.3	90.24
pglib_opf_case89_pegase	ERR	ERR	ERR	ERR	ERR
pglib_opf_case162_ieee_dtc	1465.86	12.35	35.0	98.1	97.54
pglib_opf_case240_pserc	4350.36	59.58	57.1	98.7	95.31
pglib_opf_case300_ieee	ERR	ERR	ERR	ERR	ERR
Active Power Increase					
pglib_opf_case24_ieee_rts__api	17.44	2.44	28.3	66.9	55.26
pglib_opf_case30_as__api	20.98	0.78	9.0	80.2	51.22
pglib_opf_case30_ieee__api	22.15	2.33	11.2	88.2	63.41
pglib_opf_case73_ieee_rts__api	166.12	8.05	31.9	67.7	59.17
pglib_opf_case118_ieee__api	544.96	10.36	31.4	97.4	91.40
pglib_opf_case240_pserc__api	ERR	ERR	ERR	ERR	ERR
Small Angle Difference					
pglib_opf_case3_lmbd__sad	0.14	0.07	11.2	62.3	33.33
pglib_opf_case5_pjm__sad	0.52	0.36	22.3	55.9	33.33
pglib_opf_case24_ieee_rts__sad	11.05	0.86	66.4	93.1	68.42
pglib_opf_case30_as__sad	13.52	0.82	54.1	94.3	73.17
pglib_opf_case30_ieee__sad	18.01	0.87	37.8	90.3	82.93
pglib_opf_case73_ieee_rts__sad	111.09	3.01	71.9	94.3	79.17
pglib_opf_case118_ieee__sad	412.21	6.53	69.2	98.1	95.70
pglib_opf_case162_ieee_dtc__sad	1183.13	12.71	59.6	98.1	97.54
pglib_opf_case240_pserc__sad	2502.50	56.26	79.4	98.7	95.31
pglib_opf_case300_ieee__sad	ERR	ERR	ERR	ERR	ERR
Average	571.71	9.45	40	88	73

Table 5.1 summarises the bound propagation results on the following metrics: total runtime of the algorithm, fully parallel runtime, size of θ and v domains after bound propagation (measured in percentage of the size of original domain) and number of free lines, i.e. lines where z cannot be fixed to 1 after bound propagation (measured in percentage of the total number of lines in the network).

Since there are no significant differences in bound propagation results between different modifications of the QC-OTS model, only the results that correspond to the model that includes all proposed improvements are presented in this subsection.

5.7.2 Results on the QC-OTS models

This subsection discusses the results on the QC-OTS models. The convergence tolerance on the relative difference between upper and lower bounds on the solutions of mixed-integer problems was $\epsilon = 0.01$, and the time limit was $7200s$.

We present the results for the following modifications of the QC-OTS model:

- S - simple QC-OTS model without any improvements,
- BP - model with bound propagation,
- Qtrig - model with bound propagation and improved quadratic relaxations of trigonometric functions,
- All_Impr - model that includes all improvements introduced in this chapter (the convex hull formulation for quadratic on/off constraints, bound propagation, quadratic relaxations of trigonometric functions, improved big-M calculation, LNC cuts).

'SQ' and 'M' at the end of model names stand respectively for the model with the Smart Quadratic on/off constraints described in Subsection 5.6.3 and with the big-M formulations.

Table 5.2: Running times (s)

Test Case	All_Impr	Qtrig-SQ	Qtrig-M	BP-SQ	BP-M	S-SQ	S-M
pglib_opf_case3_lmbd	0.02	0.0	0.0	0.0	0.0	0.0	0.0
pglib_opf_case5_pjm	**0.07**	0.1	0.1	0.1	0.1	0.1	0.1
pglib_opf_case30_ieee	6.32	9.3	6.5	6.8	10.5	**0.7**	0.8
pglib_opf_case89_pegase	ERR	ERR	ERR	ERR	ERR	103.4	**80.3**
pglib_opf_case162_ieee_dtc	95.47	93.5	255.3	91.4	218.6	**70.3**	77.7
pglib_opf_case240_pserc	638.03	ERR	949.3	623.8	ERR	**411.5**	1497.4
pglib_opf_case300_ieee	7200.0	7200.0	7200.0	7200.0	7200.0	7200.0	7200.0
Active Power Increase							
pglib_opf_case24_ieee_rts__api	10.01	8.0	7.6	5.6	7.7	**7.0**	11.4
pglib_opf_case30_as__api	2.77	4.0	3.2	3.2	3.1	**1.6**	**1.6**
pglib_opf_case30_ieee__api	1.07	1.1	0.9	1.1	0.9	**0.6**	0.7
pglib_opf_case73_ieee_rts__api	356.5	221.1	213.7	146.0	332.8	**10.3**	682.0
pglib_opf_case118_ieee__api	7200.03	7200.1	7200.0	7200.0	7200.0	**6.6**	25.2
pglib_opf_case240_pserc__api	ERR	ERR	387.2	ERR	550.6	**265.6**	304.2
Small Angle Difference							
pglib_opf_case3_lmbd__sad	0.02	0.0	0.0	0.0	0.0	0.0	0.0
pglib_opf_case5_pjm__sad	0.04	0.1	0.1	0.0	0.0	0.0	0.1
pglib_opf_case24_ieee_rts__sad	31.93	50.4	31.5	31.2	36.2	**29.1**	45.3
pglib_opf_case30_as__sad	**3.27**	5.3	5.6	5.9	5.3	8.0	10.2
pglib_opf_case30_ieee__sad	2.22	2.0	1.6	2.0	**1.4**	1.6	1.6
pglib_opf_case73_ieee_rts__sad	337.11	145.0	156.2	**144.8**	444.0	220.3	271.4
pglib_opf_case118_ieee__sad	528.63	698.0	757.5	1105.3	**393.8**	1615.4	649.6
pglib_opf_case162_ieee_dtc__sad	7200.01	7200.0	7200.0	7200.0	7200.0	**70.7**	95.7
pglib_opf_case240_pserc__sad	7200.02	7200.0	7200.0	7200.0	7200.0	674.1	**659.4**
pglib_opf_case300_ieee__sad	7200	7200.0	7200.0	7200.0	7200.0	7200.0	7200.0
Geometric average	21.6	24.9	31.7	25.8	27.8	**12.7**	18.4

Table 5.2 shows the running times of the branch and bound algorithm used for lower bound computation (excluding the time spent on bound propagation) in seconds. It can be seen that using the new formulations of disjunctive quadratic constraints tends to improve the runtime compared to the big-M relaxations, and model S-SQ is the fastest among all presented formulations.

Bound propagation increases the average time required to solve the MINLPs. This can be explained by the fact that the variable bounds can become very tight, which makes the models numerically more challenging. Moreover, in some cases this may cause the solver to experience numerical issues which result in the errors appearing in models BP and Qtrig.

Table 5.3: Optimality gaps (%)

Test Case	AC-OTS cost	All_Impr	Qtrig	BP	S
pglib_opf_case3_lmbd	5812.6	0.3	0.3	0.3	1.50
pglib_opf_case5_pjm	15174	1.1	1.1	1.1	1.15
pglib_opf_case30_ieee	11492	**2.8**	3.0	3.1	7.27
pglib_opf_case89_pegase	116087	ERR	ERR	ERR	**0.74**
pglib_opf_case162_ieee_dtc	119794	3.0	3.0	3.0	3.04
pglib_opf_case240_pserc	-	-	-	-	-
pglib_opf_case300_ieee	654039	-	-	-	-
Active Power Increase					
pglib_opf_case24_ieee_rts__api	119822	1.0	1.0	1.0	5.24
pglib_opf_case30_as__api	2797.5	1.1	1.1	1.1	1.23
pglib_opf_case30_ieee__api	24107.2	1.2	1.2	1.2	4.18
pglib_opf_case73_ieee_rts__api	385746	**1.2**	**1.2**	1.4	4.35
pglib_opf_case118_ieee__api	258170	12.7	12.7	12.7	13.33
pglib_opf_case240_pserc__api	-	-	-	-	-
Small Angle Difference					
pglib_opf_case3_lmbd__sad	5959.3	**0.2**	0.3	0.3	1.42
pglib_opf_case5_pjm__sad	26115.2	**0.5**	0.7	0.7	1.13
pglib_opf_case24_ieee_rts__sad	76641.1	2.8	2.6	**2.5**	3.97
pglib_opf_case30_as__sad	-	-	-	-	-
pglib_opf_case30_ieee__sad	11963.3	**0.8**	**0.8**	0.9	4.92
pglib_opf_case73_ieee_rts__sad	222401	1.2	1.2	**1.1**	2.37
pglib_opf_case118_ieee__sad	116884	2.2	2.2	2.2	2.24
pglib_opf_case162_ieee_dtc__sad	120298	3.3	3.3	3.3	3.45
pglib_opf_case240_pserc__sad	-	-	-	-	-
pglib_opf_case300_ieee__sad	-	-	-	-	-
Average		**2.22**	**2.22**	2.24	3.62

Table 5.3 presents the optimality gaps. Bound propagation significantly tightens the relaxations and thus improves the gap, with the best improvements yielded for instances pglib_case30_ieee (4.5 percentage point difference between the gaps of S and All_Impr models), pglib_case24_ieee_rts__api (4.2 percentage points), pglib_case73_ieee_rts__api (3.1 percentage points) and pglib_case30_ieee__sad (4.1 percentage point). The difference between the All_Impr, Qtrig and BP models, however, is small. Models SQ and M have similar optimality gaps and are not separated in this table.

For some instances a tighter formulation yields larger gaps than a less strong model. However, in these cases the differences in gaps are always below the solver tolerance $\epsilon = 0.01$.

5.7.3 Comparison with MISOCP

In Table 5.4 we compare the gaps yielded by model All_Impr and the MISOCP model [113] on standard IEEE instances and NESTA [39] - v0.3.0 instances with congested conditions. The time limit for this table was set to $3600s$. The set of instances here is identical to that used for experiments with MISOCP [113]. On the IEEE test cases, there is no significant difference in the results since both model yield a gap that is close to zero. For the more challenging NESTA Active Power Increase instances, using the All_Impr formulation improves the gap for the nesta_case118_ieee__api instance by 3.13 percentage points and for the nesta_case_189_edin__api instance by 1.46 percentage points and closes the gap on all other instances where the gap of the MISOCP approach was larger than 1% (nesta_case3_lmbd__api and nesta_case6_ww__api).

Table 5.4: Comparing results with the MISOCP model [113]

IEEE standard			
Test Case	AC-OTS cost	Gap - All_Impr (%)	Gap - MISOCP (%)
6ww	3128.8	0.01	0.01
9	5296.7	0.01	0.00
9Q	5296.7	**0.00**	0.04
14	8082.3	0.10	**0.01**
ieee30	8907.7	0.08	**0.02**
30	573.9	0.09	**0.03**
30Q	573.9	**0.09**	0.13
39	41857.2	0.01	0.01
57	41778.4	0.27	**0.08**
118	130355	0.79	**0.17**
300	719807	0.18	**0.10**
Average		0.15	**0.05**

NESTA Active Power Increase			
Test Case	AC-OTS cost	Gap - All_Impr (%)	Gap - MISOCP (%)
nesta_case3_lmbd__api	367	**0.62**	1.17
nesta_case4gs__api	767	0.00	0.00
nesta_case5_pjm__api	2987	0.02	0.02
nesta_case6_ww__api	252	**0.54**	1.05
nesta_case9_wscc__api	656	0.00	0.00
nesta_case14_ieee__api	321	**0.31**	0.41
nesta_case29_edin__api	295160	**0.21**	0.33
nesta_case30_as__api	553	**0.31**	0.34
nesta_case30_ieee__api	409	0.18	**0.15**
nesta_case30_fsr__api	204	0.14	**0.03**
nesta_case39_epri__api	7359	**0.41**	0.70
nesta_case57_ieee__api	1429	0.10	**0.09**
nesta_case118_ieee__api	6018	**4.37**	7.50
nesta_case162_ieee_dtc__api	6018	**0.36**	0.60
nesta_case189_edin__api	1947	**4.12**	5.58
nesta_case300_ieee__api	22825	1.04	**0.61**
Average		**0.80**	1.16

Note that All_Impr and MISOCP have a different heuristic for computing the upper bounds and use different algorithms and hardware to compute the lower bounds.

5.8 Conclusion

In this chapter we have proposed an explicit formulation of a one-dimensional quadratic disjunctive constraint which leads to tighter continuous relaxations than the standard big-M formulation and does not require adding new variables into the model. This result was applied to the Convex Quadratic relaxation of the Optimal Transmission Switching problem. Computational experiments showed that the new convex hull formulation leads to an improvement in average solution times. We strengthened the QC relaxation by applying bound propagation, introducing tighter quadratic relaxations of the trigonometric terms and adding variables representing directions of power flows to the model. The optimality gaps were significantly reduced with the best improvement yielded by bound propagation.

Chapter 6

Conclusion

6.1 Main Results

In this thesis we have studied global optimisation methods with provable optimality guarantees, aiming at solving large scale problems in energy systems which are characterised by non-convexity and incorporation of discrete elements. The focus of the work has been on improving the computational efficiency and the quality of guarantees provided. The proposed methods have been applied to the Optimal Power Flow problem and one of its mixed-integer extensions known as Optimal Transmission Switching. Since many other problems in energy systems are based on Optimal Power Flow, the techniques introduced in the thesis can be applied to them with minor modifications.

Some problems possess convexity-like properties and can be solved to global optimality by local optimisation methods that converge to the local optimum in the general case. To utilise such properties for providing global optimality guarantees, they need to be algorithmically verifiable and proven theoretically.

We examined conditions under which every Karush-Kuhn-Tucker point is a global optimum (i.e. a problem is KT-invex). We considered nonconvex maximisation problems with concave objective functions and introduced subproblems minimising the objective function over subsets of the boundary of the feasible set. We defined boundary-invex problems by introducing algorithmically verifiable requirements on the stationary points of the minimisation subproblems. Boundary-invexity was proven to be necessary and sufficient for KT-invexity of problems with two degrees of freedom and necessary for KT-invexity for all continuous nonconvex problems that satisfy mild nondegeneracy conditions. This contribution is the first step in bridging the gap between invexity theory and practical applications.

Applying the new conditions to the Optimal Power Flow problem, we showed that it is boundary-invex on networks consisting of two nodes with two generators connected by a line under mild assumptions on the parameters. Since on such networks OPF has two degrees of freedom, boundary-invexity implies KT-invexity.

Convex relaxations provide a different approach for proving global optimality. By computing the relative difference between the lower bound yielded by the relaxation and the upper bound obtained from the local optimum of the nonconvex problem, one can determine how far from the global optimal value the value at a local solution is. In this work we studied semidefinite programming problems which are commonly used as convex relaxations of nonconvex problems, including Optimal Power Flow, and proposed an improved polynomial relaxation and a new linear cut generation algorithm.

Many problems in energy systems contain discrete elements and are modelled as mixed-integer nonlinear programs. The formulation of a problem affects the performance of solution algorithms. In the case of MINLPs, we are interested in finding formulations that yield tight convex relaxations but have a reasonable number of variables and constraints. We considered a quadratic nonisotone on/off constraint and derived a formulation of the convex hull of its feasible set in the space of original variables.

Applying this result to the Quadratic Convex relaxation of the Optimal Transmission Switching problem improves the average solution time. In order to decrease the gap yielded by the relaxation, we introduced several improvements such as bound propagation, smarter big-M constant calculation and on/off lifted nonlinear cuts. Strengthening the model closes the gap on 5 out of 23 instances from the PGLib benchmark.

The results presented in this thesis improve global optimisation methods for nonconvex continuous and mixed-integer problems and extend the capabilities of optimisation in the practical area of energy systems. The presented methods can either be applied immediately to find reliable solutions of real life problems or bring the state-of-the-art optimisation technology closer to the point where this is possible. The theoretical findings of the thesis expand the knowledge in the area of optimisation and are applicable to a wider range of applications outside of energy systems.

6.2 Future Work

This thesis has shown several interesting directions that can be further researched.

Extending the applicability of boundary-invexity The sufficiency proof presented in Chapter 3 is valid only for problems with two degrees of freedom and is a first step in extending the reach of interior-point methods to nonconvex problems. If similar properties can be proven for problems with an arbitrary number of degrees of freedom, this would significantly extend the applicability of the concept of invexity. Invexity of equality constrained problems could be studied, since in many cases obtaining an equivalent inequality constrained formulation of a given problem is not straightforward.

Verifying the invexity of a problem using our proposed method is NP-hard in general even in the case of two degrees of freedom. If generalised to problems with n degrees of freedom, the number of subproblems that need to be solved to detect boundary-invexity will be larger, making the verification more computationally expensive. However, as we briefly mentioned in Chapter 3, in some special cases this task becomes much easier: for example, for quadratically constrained quadratic problems (with some mild assumptions) the verification can be performed in polynomial time. Future research would identify other special cases where invexity can be detected with less computational effort. Methods for efficiently solving the problems used to define the boundary-invexity condition can be studied, making use of the convexity of the objective function as well as any additional information on constraints.

It would be interesting to investigate the invexity of other optimisation problems, particularly those that exhibit invex-like behaviour in practice (that is, where KKT points tend to be global optima). Future work could also study invex relaxations of non-convex, non-invex problems: since invexity is a weaker requirement than convexity, invex relaxations might yield better bounds.

Studying connected and disconnected sets Since the invexity proofs rely on the connectedness of the feasible region, deriving conditions under which a nonconvex set is connected is another important step for increasing the practical relevance of this research. Developing specialised algorithms for problems with disconnected feasible sets could be another promising research direction, especially since it has been observed that for the Optimal Power Flow problem multiple local minima often exist due to the disconnectedness of the feasible region [139].

Improving branching rules Chapter 3 provides new perspectives on the connection between the structure of the problem and existence of multiple local optima, which can be used for improving branching rules of spatial branch-and-bound algorithms. The new rules would aim for obtaining invex or "more invex" subregions by dividing the sets at the points of violation of invexity, which could result in better convergence.

Better linearisation techniques Generating linear outer approximations of nonlinear sets is an actively researched area due to the usefulness of separating hyperplanes for various optimisation algorithms. The applicability of the deepest valid cut described in Chapter 4 in different contexts could be investigated. However, this approach can be computationally expensive since it requires solving a large number of projection subproblems. Future research would develop algorithms for more efficient linear cut generation for convex sets described by possibly non-convex constraints, building upon the contributions of this thesis and the already existing rich literature on linearisation techniques.

Utilising special structure It was noted in Chapter 4 that the tree decomposition approach benefits from the fact that the graphs representing electric networks tend to be planar or near-planar, which leads to smaller tree decomposition bags. It is possible that this property could be further utilised in power system optimisation methods - see, for example, the report by Erikson et al. [57] on optimisation algorithms for planar graphs.

In Chapter 4 we observed that the Second Order Cone constraints tend to be active at an optimal solution of Optimal Power Flow, and when this holds and the vertices are contained in a tree decomposition bag of size 3, the matrix corresponding to the bag has a unique SDP completion. In other words, when a constraint corresponding to a minor of size 2 of a three dimensional Hermittian matrix is active, the SDP system for this matrix has a unique, easily computable solution. This observation can potentially be used in other applciations and in general SDP programming.

SDP cuts for mixed-integer problems An interesting extension of the linear cut generation approach introduced in Chapter 4 is to apply it in the mixed-integer case. Since the nonlinear SDP constraints together with discrete variables cause significant computational difficulties, a linear programming reformulation is promising.

Avoiding nondifferentiability in perspective formulations Perspective formulations employed in Chapter 5 have the problem of being nondifferentiable when the binary variable is equal to zero. In this work, we used linear outer approximations of the convex hull to avoid this issue, but better options could be explored. One interesting approach that tackles the nondifferentiability is the "project and lift" approach introduced by Frangioni et al. [62]. In their work, the authors study the case when the "off" state is represented by a single point. Future research would investigate the possibility for extending these results to the types of on/off constraints considered in this thesis and applying them to the convex hull formulation for non-monotone quadratic disjunctive constraints presented in Chapter 5.

Improving convex relaxations of Optimal Power Flow The Quadratic Convex relaxation studied in Chapter 5 captures aspects of nonlinearity of the OPF problem that are different from those accounted for by the Semidefinite Programming relaxation. This suggests that combining QC-OTS with the disjunctive versions of the linear cuts proposed in Chapter 4 will produce formulations with the combined bounding strength of the two approaches. Other research directions include seeking ways to further tighten convex relaxations of Optimal Power Flow and improve their computational efficiency.

Publications

Conferences

- K. Bestuzheva, H. Hijazi, C. Coffrin. Convex Relaxations of Quadratic On/Off Constraints and Applications to Optimal Transmission Switching. The 24th ASOR National Conference, Canberra, Australia, November 2016.

- K. Bestuzheva, H. Hijazi. Kuhn-Tucker Invexity of Non-convex Optimisation Problems with Two Degrees of Freedom. Global Optimization Conference, College Station, USA, March-April 2017.

- K. Bestuzheva, H. Hijazi. Global Optimization for Alternating Current Optimal Power Flow. International Symposium on Mathematical Programming, Bordeaux, France, July 2018.

Journals

- K. Bestuzheva, H. Hijazi. Invex Optimization Revisited. Journal of Global Optimization, April 2018.

- K. Bestuzheva, H. Hijazi, C. Coffrin. Convex Relaxations for Quadratic On/Off Constraints and Applications to Optimal Transmission Switching. Accepted for publication in INFORMS Journal on Computing in March 2019.

Appendix A

Improved On/Off Relaxations of Nonlinear Terms

Here we give the full formulation of the improved relaxations of on/off trigonometric constraints.

A.1 Sine Constraint

We begin with the sine relaxation. Let the variable sn represent the convex relaxation of $\sin(\theta)$ with $-\pi/2 < \boldsymbol{\theta}^l \leqslant \theta \leqslant \boldsymbol{\theta}^u < \pi/2$ and the binary variable be denoted as z. The formulation depends on the signs of $\boldsymbol{\theta}^l$ and $\boldsymbol{\theta}^u$.

Unknown sign: $\boldsymbol{\theta}^l < 0$ and $\boldsymbol{\theta}^u > 0$. Since the new quadratic relaxations require the knowledge of the sign of θ, in this case the polyhedral formulation described in Subsection 2.6.1 of Chapter 2 has to be used. First, let us write the linear relaxation of sine in the "on" state with asymmetrical bounds:

$$sn \leqslant \cos\left(\frac{\boldsymbol{\theta}^m}{2}\right)\left(\theta - \frac{\boldsymbol{\theta}^m}{2}\right) + \sin\left(\frac{\boldsymbol{\theta}^m}{2}\right),$$

$$sn \geqslant \cos\left(\frac{\boldsymbol{\theta}^m}{2}\right)\left(\theta + \frac{\boldsymbol{\theta}^m}{2}\right) - \sin\left(\frac{\boldsymbol{\theta}^m}{2}\right),$$

where $\boldsymbol{\theta}^m = \max(-\boldsymbol{\theta}^l, \boldsymbol{\theta}^u)$. Remembering that $sn = 0$ and $\boldsymbol{M}^l \leqslant \theta \leqslant \boldsymbol{M}^u$ when $z = 0$, if we let $\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}} = \boldsymbol{\theta}^m/2$ and apply formulation (2.31)) for disjunctive linear on/off constraints, we get:

$$sn - \cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\theta \leqslant z(\sin(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}}) - \cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}}) - (1-z)\cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\boldsymbol{M}^l,$$

$$\cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\theta - sn \leqslant z(\sin(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}}) - \cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}}) + (1-z)\cos(\boldsymbol{\theta}^{\boldsymbol{m}}_{\boldsymbol{1/2}})\boldsymbol{M}^u.$$

Nonnegative angle difference: $\boldsymbol{\theta}^l \geqslant 0$. In the case when θ is known to be nonnegative, the new quadratic outer approximation introduced in Section 5.5 can be applied to obtain the upper bound on sn. In the "on" state, it has the form

$$sn \leqslant Q_{sin}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon),$$

where Q_f is a quadratic function whose graph passes through points $(\boldsymbol{x}^1, f(\boldsymbol{x}^1))$, $(\boldsymbol{x}^2, f(\boldsymbol{x}^2))$ and $(\boldsymbol{x}^3, f(\boldsymbol{x}^3))$ and $\epsilon < \pi/2 - \boldsymbol{\theta}^u$.

The lower bound is given by the following linear constraint:

$$sn \geqslant \frac{\sin(\boldsymbol{\theta}^l) - \sin(\boldsymbol{\theta}^u)}{\boldsymbol{\theta}^l - \boldsymbol{\theta}^u}(\theta - \boldsymbol{\theta}^l) + \sin(\boldsymbol{\theta}^l).$$

To obtain the disjunctive version of the above linear inequality, formulation (2.31) is used. For the quadratic upper bound, one option is to apply a big-M formulation:

$$sn \leqslant Q_{sin}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon) - (1 - z)\boldsymbol{M}^+_{sin},$$

$$\boldsymbol{M}^+_{sin} = \min_{\theta \in [\boldsymbol{M}^l, \boldsymbol{M}^u]} Q_{sin}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon).$$

Another option is to use cuts based on the convex hull formulation introduced in Section 5.4. This is done by adding linear outer approximations of the following nonlinear constraint:

$$\theta_{ij} - \boldsymbol{M}^l\,(1 - z) + \boldsymbol{\rho} z \geqslant -\sqrt{\frac{sn\ z + \delta z^2}{\boldsymbol{a}}},$$
$$\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c},$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are the coefficients of $Q_{sin}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon)$.

Nonpositive angle difference: $\boldsymbol{\theta}^u \leqslant 0$. This case is symmetric to the previous one. For an active line, the upper and lower bounds are

$$sn \leqslant \frac{\sin(\boldsymbol{\theta}^l) - \sin(\boldsymbol{\theta}^u)}{\boldsymbol{\theta}^l - \boldsymbol{\theta}^u}(\theta - \boldsymbol{\theta}^l) + \sin(\boldsymbol{\theta}^l),$$

$$sn \geqslant Q_{sin}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon),$$

where $\epsilon < \boldsymbol{\theta}^l + \pi/2$. The big-M formulation of the on/off quadratic lower bound is given by:

$$sn \geqslant Q_{sin}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon) - (1 - z)\boldsymbol{M}^-_{sin},$$

$$\boldsymbol{M}^-_{sin} = \max_{\theta \in [\boldsymbol{M}^l, \boldsymbol{M}^u]} Q_{sin}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon),$$

and the convex hull is described by:

$$\begin{aligned}
&\theta - \boldsymbol{M}^u (1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{sn\ z + \delta z^2}{\boldsymbol{a}}},\\
&\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c},
\end{aligned}$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are the coefficients of $Q_{sin}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon)$.

A.2 Cosine Constraint

Now we move on to the cosine relaxation. Regardless of the signs of $\boldsymbol{\theta}^l$ and $\boldsymbol{\theta}^u$, the lower bound on cs is given by the following linear inequality:

$$cs \geqslant \frac{\cos(\boldsymbol{\theta}^l) - \cos(\boldsymbol{\theta}^u)}{\boldsymbol{\theta}^l - \boldsymbol{\theta}^u}(\theta - \boldsymbol{\theta}^l) + \cos(\boldsymbol{\theta}^l)$$

and its disjunctive version is obtained by applying formulation (2.31).

Unknown sign: $\boldsymbol{\theta}^l < 0$ and $\boldsymbol{\theta}^u > 0$. In this case the third point defining the quadratic outer approximation is chosen in the middle and $\boldsymbol{\theta}^m$ replaces $\boldsymbol{\theta}^l$ and $\boldsymbol{\theta}^u$:

$$cs \leqslant Q_{cos}(-\boldsymbol{\theta}^m, \boldsymbol{\theta}^m, 0).$$

This constraint is equivalent to the quadratic outer approximation introduced by Hijazi et al. [100]. The big-M formulation of the disjunctive constraint has the form:

$$cs \leqslant Q_{cos}(-\boldsymbol{\theta}^m, \boldsymbol{\theta}^m, 0) - (1-z)\boldsymbol{M}_{cos},$$

$$\boldsymbol{M}_{cos} = \min_{\theta \in [\boldsymbol{M}^l, \boldsymbol{M}^u]} Q_{cos}(-\boldsymbol{\theta}^m, \boldsymbol{\theta}^m, 0).$$

Since in all other cases only the choice of the quadratic outer approximating function is different, we will omit the subsequent big-M constraints.

For the convex hull we have:

$$\begin{aligned}
&\theta - \boldsymbol{M}^u (1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{cs\ z + \delta z^2}{\boldsymbol{a}}},\\
&\theta - \boldsymbol{M}^l (1-z) + \boldsymbol{\rho} z \geqslant -\sqrt{\frac{cs\ z + \delta z^2}{\boldsymbol{a}}},\\
&\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c},
\end{aligned}$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are the coefficients of $Q_{cos}(\boldsymbol{\theta}^m, \boldsymbol{\theta}^m, 0)$.

Nonnegative angle difference: $\boldsymbol{\theta}^l \geqslant 0$. Similarly to the sine relaxation, the new asymmetric quadratic relaxations can be used when the sign of the argument is known. Thus the constraint in the "on" state is written as:

$$cs \leqslant Q_{cos}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon),$$

and the convex hull of the disjunction is given by:

$$\begin{aligned}
&\theta - \boldsymbol{M}^u\,(1-z) + \boldsymbol{\rho} z \leqslant \sqrt{\frac{cs\ z + \delta z^2}{\boldsymbol{a}}},\\
&\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c},
\end{aligned}$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are the coefficients of $Q_{cos}(\boldsymbol{\theta}^u, \boldsymbol{\theta}^l, \boldsymbol{\theta}^l - \epsilon)$.

Nonpositive angle difference: $\boldsymbol{\theta}^u \leqslant 0$. When $z = 1$, we have:

$$cs \leqslant Q_{cos}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon),$$

and the on/off version of this constraint is characterised by the following inequality:

$$\begin{aligned}
&\theta - \boldsymbol{M}^l\,(1-z) + \boldsymbol{\rho} z \geqslant -\sqrt{\frac{cs\ z + \delta z^2}{\boldsymbol{a}}},\\
&\rho = \frac{\boldsymbol{b}}{2\boldsymbol{a}},\ \delta = \boldsymbol{a}\boldsymbol{\rho}^2 - \boldsymbol{c},
\end{aligned}$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are the coefficients of $Q_{cos}(\boldsymbol{\theta}^l, \boldsymbol{\theta}^u, \boldsymbol{\theta}^u + \epsilon)$.

Bibliography

[1] Abbena, E., Salamon, S., and Gray, A. *Modern differential geometry of curves and surfaces with Mathematica.* CRC Press, 2006.

[2] Aktürk, M. S., Atamtürk, A., and Gürel, S. A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Operations Research Letters 37*, 3 (2009), 187–191.

[3] Alizadeh, F. Optimization over the positive-definite cone: interior point methods and combinatorial applications. *Advances in Optimization and Parallel Computing* (1992).

[4] ApS, M. *MOSEK Fusion API for C++ 8.1.0.56*, 2018.

[5] Avriel, M. *Nonlinear programming: analysis and methods.* Courier Corporation, 2003.

[6] Avriel, M., and Zang, I. Generalized arcwise-connected functions and characterizations of local-global minimum properties. *Journal of Optimization Theory and Applications 32*, 4 (1980), 407–425.

[7] Bai, X., and Wei, H. A semidefinite programming method with graph partitioning technique for Optimal Power Flow problems. *International Journal of Electrical Power & Energy Systems 33*, 7 (2011), 1309–1314.

[8] Bai, X., Wei, H., Fujisawa, K., and Wang, Y. Semidefinite programming for Optimal Power Flow problems. *International Journal of Electrical Power & Energy Systems 30*, 6-7 (2008), 383–392.

[9] Balas, E. Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. *SIAM Journal on Algebraic Discrete Methods 6*, 3 (1985), 466–486.

[10] Balas, E. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Mathematics 89*, 1-3 (1998), 3–44.

[11] Balas, E., Ceria, S., and Cornuéjols, G. A lift-and-project cutting plane algorithm for mixed 0–1 programs. *Mathematical Programming 58*, 1-3 (1993), 295–324.

[12] Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research 9*, Mar (2008), 485–516.

[13] Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., and Vance, P. H. Branch-and-price: Column generation for solving huge integer programs. *Operations Research 46*, 3 (1998), 316–329.

[14] Barrows, C., and Blumsack, S. Transmission switching in the RTS-96 test system. *IEEE Transactions on Power Systems 27*, 2 (May 2012), 1134–1135.

[15] Barrows, C., Blumsack, S., and Bent, R. Computationally efficient optimal transmission switching: Solution space reduction. In *Power and Energy Society General Meeting, 2012 IEEE* (July 2012), pp. 1–8.

[16] Barrows, C., Blumsack, S., and Hines, P. Correcting optimal transmission switching for AC power flows. In *2014 IEEE PES General Meeting — Conference Exposition* (July 2014), pp. 1–5.

[17] Belotti, P., Bonami, P., Fischetti, M., Lodi, A., Monaci, M., Nogales-Gómez, A., and Salvagnin, D. On handling indicator constraints in mixed integer programming. *Computational Optimization and Applications 65*, 3 (2016), 1–22.

[18] Ben-Israel, A., and Mond, B. What is invexity? *The ANZIAM Journal 28*, 1 (1986), 1–9.

[19] Ben-Tal, A. On generalized means and generalized convex functions. *Journal of Optimization Theory and Applications 21*, 1 (1977), 1–13.

[20] Benders, J. F. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik 4*, 1 (1962), 238–252.

[21] Bertsimas, D., and Tsitsiklis, J. N. *Introduction to linear optimization*, vol. 6. Athena Scientific Belmont, MA, 1997.

[22] Bienstock, D., and Munoz, G. On linear relaxations of OPF problems. *arXiv preprint arXiv:1411.1120* (2014).

[23] Bonami, P., Cornuéjols, G., Lodi, A., and Margot, F. A feasibility pump for mixed integer nonlinear programs. *Mathematical Programming 119*, 2 (2009), 331–352.

[24] Bonami, P., Kılınç, M., and Linderoth, J. Algorithms and software for convex mixed integer nonlinear programs. In *Mixed Integer Nonlinear Programming*. Springer, 2012, pp. 1–39.

[25] Boyd, S., and Vandenberghe, L. *Convex optimization*. Cambridge University Press, 2004.

[26] Browne, E. T. *Introduction to the Theory of Determinants and Matrices.* University of North Carolina, 1958.

[27] Burchett, R., Happ, H., and Wirgau, K. Large scale Optimal Power Flow. *IEEE Transactions on Power Apparatus and Systems*, 10 (1982), 3722–3732.

[28] Burer, S. Optimizing a polyhedral-semidefinite relaxation of completely positive programs. *Mathematical Programming Computation 2*, 1 (2010), 1–19.

[29] Burer, S., and Monteiro, R. D. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming 95*, 2 (2003), 329–357.

[30] Byrd, R. H., Gould, N. I., Nocedal, J., and Waltz, R. A. An algorithm for nonlinear optimization using linear programming and equality constrained subproblems. *Mathematical Programming 100*, 1 (2003), 27–48.

[31] Capitanescu, F., Ramos, J. M., Panciatici, P., Kirschen, D., Marcolini, A. M., Platbrood, L., and Wehenkel, L. State-of-the-art, challenges, and future trends in security constrained Optimal Power Flow. *Electric Power Systems Research 81*, 8 (2011), 1731–1741.

[32] Carpentier, J. Contribution to the economic dispatch problem. *Bulletin de la Societe Francoise des Electriciens 3*, 8 (1962), 431–447.

[33] Castagnoli, E., and Mazzoleni, P. Towards a unified type of concavity. *Journal of Information and Optimization Sciences 10*, 1 (1989), 225–240.

[34] Ceria, S., and Soares, J. Convex programming for disjunctive convex optimization. *Mathematical Programming 86* (12 1999), 595–614.

[35] Chen, C., Atamtürk, A., and Oren, S. S. A spatial branch-and-cut method for nonconvex QCQP with bounded complex variables. *Mathematical Programming 165*, 2 (2017), 549–577.

[36] Chen, S.-D., and Chen, J.-F. A new algorithm based on the Newton-Raphson approach for real-time emission dispatch. *Electric Power Systems Research 40*, 2 (1997), 137–141.

[37] Cheney, E. W., and Goldstein, A. A. Newton's method for convex programming and Tchebycheff approximation. *Numerische Mathematik 1*, 1 (1959), 253–268.

[38] Chor, B., and Sudan, M. A geometric approach to betweenness. *SIAM Journal on Discrete Mathematics 11*, 4 (1998), 511–523.

[39] Coffrin, C., Gordon, D., and Scott, P. NESTA, The Nicta Energy System Test Case Archive. *CoRR abs/1411.0359* (2014).

[40] Coffrin, C., Hijazi, H., Lehmann, K., and Van Hentenryck, P. Primal and dual bounds for Optimal Transmission Switching. *Power Systems Computation Conference (PSCC)* (08 2014), 1–8.

[41] Coffrin, C., Hijazi, H., and Van Hentenryck, P. Network flow and copper plate relaxations for AC transmission systems. In *Power Systems Computation Conference (PSCC), 2016* (2016), IEEE, pp. 1–8.

[42] Coffrin, C., Hijazi, H. L., and Hentenryck, P. V. Strengthening the SDP relaxation of AC power flows with convex envelopes, bound tightening, and lifted nonlinear cuts. *CoRR abs/1512.04644* (2015).

[43] Coffrin, C., Hijazi, H. L., and Van Hentenryck, P. Strengthening convex relaxations with bound tightening for power network optimization. In *International Conference on Principles and Practice of Constraint Programming* (2015), Springer, pp. 39–57.

[44] Coffrin, C., and Van Hentenryck, P. A linear-programming approximation of AC power flows. *INFORMS Journal on Computing 26*, 4 (2014), 718–734.

[45] Coffrin, C., Van Hentenryck, P., and Bent, R. Accurate load and generation scheduling for linearized DC models with contingencies. In *Power and Energy Society General Meeting, 2012 IEEE* (2012), IEEE, pp. 1–8.

[46] Conn, A. R., Gould, N. I., and Toint, P. L. *Trust region methods*, vol. 1. SIAM, 2000.

[47] Craven, B. Invex functions and constrained local minima. *Bulletin of the Australian Mathematical Society 24*, 03 (1981), 357–366.

[48] Craven, B. Global invexity and duality in mathematical programming. *Asia-Pacific Journal of Operational Research 19*, 2 (2002), 169.

[49] Craven, B., and Glover, B. Invex functions and duality. *Journal of the Australian Mathematical Society (Series A) 39*, 01 (1985), 1–20.

[50] Dakin, R. J. A tree-search algorithm for mixed integer programming problems. *The Computer Journal 8*, 3 (1965), 250–255.

[51] Dantzig, G., Fulkerson, R., and Johnson, S. Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America 2*, 4 (1954), 393–410.

[52] Dey, S. S., and Molinaro, M. Theoretical challenges towards cutting-plane selection. *Mathematical Programming 170*, 1 (2018), 237–266.

[53] Dommel, H. W., and Tinney, W. F. Optimal Power Flow solutions. *IEEE Transactions on Power Apparatus and Systems*, 10 (1968), 1866–1876.

[54] Duran, M. A., and Grossmann, I. E. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming 36*, 3 (1986), 307–339.

[55] El Ghaoui, L., and Niculescu, S.-l. *Advances in linear matrix inequality methods in control.* SIAM, 2000.

[56] Eltved, A., Dahl, J., and Andersen, M. S. On the robustness and scalability of semidefinite relaxation for optimal power flow problems. *Optimization and Engineering* (2018), 1–18.

[57] Erickson, J., Klein, P. N., Marx, D., and Mathieu, C. Algorithms for optimization problems in planar graphs (Dagstuhl seminar 16221). In *Dagstuhl Reports* (2016), vol. 6, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[58] Fisher, E., O'Neill, R., and Ferris, M. Optimal transmission switching. *IEEE Transactions on Power Systems 23*, 3 (2008), 1346–1355.

[59] Fisher, E. B., O'Neill, R. P., and Ferris, M. C. Optimal Transmission Switching. *IEEE Transactions on Power Systems 23*, 3 (2008), 1346–1355.

[60] Fletcher, R., and de la Maza, E. S. Nonlinear programming and nonsmooth optimization by successive linear programming. *Mathematical Programming 43*, 1-3 (1989), 235–256.

[61] Fletcher, R., and Leyffer, S. Solving mixed integer nonlinear programs by outer approximation. *Mathematical Programming 66*, 1-3 (1994), 327–349.

[62] Frangioni, A., Furini, F., and Gentile, C. Approximated perspective relaxations: a project and lift approach. *Computational Optimization and Applications 63*, 3 (2016), 705–735.

[63] Frangioni, A., and Gentile, C. Perspective cuts for a class of convex 0–1 mixed integer programs. *Mathematical Programming 106*, 2 (2006), 225–236.

[64] Frangioni, A., Gentile, C., and Lacalandra, F. Tighter approximated MILP formulations for unit commitment problems. *IEEE Transactions on Power Systems 24*, 1 (2009), 105–113.

[65] Frank, S., and Rebennack, S. An introduction to Optimal Power Flow: Theory, formulation, and examples. *IIE Transactions 48*, 12 (2016), 1172–1197.

[66] Frieze, A., and Jerrum, M. Improved approximation algorithms for MAX k-CUT and MAX BISECTION. *Algorithmica 18*, 1 (1997), 67–81.

[67] Fujie, T., and Kojima, M. Semidefinite programming relaxation for nonconvex quadratic programs. *Journal of Global Optimization 10*, 4 (1997), 367–380.

[68] Fukuda, E. H., and Lourenço, B. F. Exact augmented Lagrangian functions for nonlinear semidefinite programming. *Computational Optimization and Applications 71*, 2 (2018), 457–482.

[69] Fukuda, M., Kojima, M., Murota, K., and Nakata, K. Exploiting sparsity in semidefinite programming via matrix completion I: General framework. *SIAM Journal on Optimization 11*, 3 (2001), 647–674.

[70] Fuller, J., Ramasra, R., and Cha, A. Fast heuristics for transmission-line switching. *IEEE Transactions on Power Systems 27*, 3 (2012), 1377–1386.

[71] Fuller, J. D., Ramasra, R., and Cha, A. Fast heuristics for transmission-line switching. *IEEE Transactions on Power Systems 27*, 3 (2012), 1377–1386.

[72] Gabay, D., and Mercier, B. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications 2*, 1 (1976), 17–40.

[73] Geoffrion, A. M. Generalized Benders decomposition. *Journal of Optimization Theory and Applications 10*, 4 (1972), 237–260.

[74] George, A., Gilbert, J. R., and Liu, J. W. *Graph theory and sparse matrix computation*, vol. 56. Springer Science & Business Media, 2012.

[75] Ghaddar, B., Marecek, J., and Mevissen, M. Optimal Power Flow as a polynomial optimization problem. *IEEE Transactions on Power Systems 31*, 1 (2016), 539–546.

[76] Glowinski, R., and Marroco, A. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *Revue Française d'Automatique, Informatique, Recherche Opérationnelle. Analyse Numérique 9*, R2 (1975), 41–76.

[77] Goemans, M. X., and Williamson, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM) 42*, 6 (1995), 1115–1145.

[78] Goemans, M. X., and Williamson, D. P. Approximation algorithms for max-3-cut and other problems via complex semidefinite programming. *Journal of Computer and System Sciences 68*, 2 (2004), 442–470.

[79] Goldfarb, D. The simplex method for conic programming. *CORC Report 5* (2002).

[80] Gomory, R. E. An algorithm for integer solutions to linear programs. *Recent Advances in Mathematical Programming 64* (1963), 260–302.

[81] Gomory, R. E., et al. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society 64*, 5 (1958), 275–278.

[82] Gopalakrishnan, A., Raghunathan, A. U., Nikovski, D., and Biegler, L. T. Global optimization of Optimal Power Flow using a branch & bound algorithm. In *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (2012), IEEE, pp. 609–616.

[83] Granville, S. Optimal reactive dispatch through interior point methods. *IEEE Transactions on Power Systems 9*, 1 (1994), 136–146.

[84] Grone, R., Johnson, C. R., Sá, E. M., and Wolkowicz, H. Positive definite completions of partial Hermitian matrices. *Linear Algebra and Its Applications 58* (1984), 109–124.

[85] Grossmann, I. E. Review of nonlinear mixed-integer and disjunctive programming techniques. *Optimization and Engineering 3*, 3 (2002), 227–252.

[86] Grossmann, I. E., and Lee, S. Generalized convex disjunctive programming: Nonlinear convex hull relaxation. *Computational Optimization and Applications 26*, 1 (2003), 83–100.

[87] Günlük, O., and Linderoth, J. Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Mathematical Programming 124*, 1-2 (2010), 183–205.

[88] Gupta, O. K., and Ravindran, A. Branch and bound experiments in convex nonlinear integer programming. *Management Science 31*, 12 (1985), 1533–1546.

[89] Hammerl, T., Musliu, N., and Schafhauser, W. Metaheuristic algorithms and tree decomposition. In *Springer Handbook of Computational Intelligence*. Springer, 2015, pp. 1255–1270.

[90] Hamzeei, M., and Luedtke, J. Linearization-based algorithms for mixed-integer nonlinear programs with convex continuous relaxation. *Journal of Global Optimization 59*, 2-3 (2014), 343–365.

[91] Hanson, M. A. On sufficiency of the Kuhn-Tucker conditions. *Journal of Mathematical Analysis and Applications 80*, 2 (1981), 545–550.

[92] Hedman, K., Ferris, M., O'Neill, R., Fisher, E., and Oren, S. Co-optimization of generation unit commitment and transmission switching with n-1 reliability. In *Power and Energy Society General Meeting, 2010 IEEE* (July 2010), pp. 1–1.

[93] Hedman, K. W., O'Neill, R. P., Fisher, E. B., and Oren, S. S. Optimal transmission switching – sensitivity analysis and extensions. *IEEE Transactions on Power Systems 23*, 3 (2008), 1469–1479.

[94] Hedman, K. W., O'Neill, R. P., Fisher, E. B., and Oren, S. S. Optimal transmission switching with contingency analysis. *IEEE Transactions on Power Systems 24*, 3 (2009), 1577–1586.

[95] Helmberg, C., Overton, M. L., and Rendl, F. The spectral bundle method with second-order information. *Optimization Methods and Software 29*, 4 (2014), 855–876.

[96] Helmberg, C., and Rendl, F. A spectral bundle method for semidefinite programming. *SIAM Journal on Optimization 10*, 3 (2000), 673–696.

[97] Henrion, D., and Malick, J. Projection methods in conic optimization. In *Handbook on Semidefinite, Conic and Polynomial Optimization*. Springer, 2012, pp. 565–600.

[98] Hijazi, H., Bonami, P., Cornujols, G., and Ouorou, A. Mixed-integer nonlinear programs featuring on/off constraints. *Computational Optimization and Applications 52*, 2 (2012), 537–558.

[99] Hijazi, H., Coffrin, C., and Van Hentenryck, P. Polynomial SDP cuts for Optimal Power Flow. In *Power Systems Computation Conference (PSCC), 2016* (2016), IEEE, pp. 1–7.

[100] Hijazi, H., Coffrin, C., and Van Hentenryck, P. Convex quadratic relaxations for mixed-integer nonlinear programs in power systems. *Mathematical Programming Computation 9*, 3 (2017), 321–367.

[101] Hijazi, H. L., Bonami, P., and Ouorou, A. A note on linear on/off constraints. *Australian National University technical report* (2014).

[102] Jabr, R. A. Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems 21*, 3 (2006), 1458–1459.

[103] Jeroslow, R. G. Representability in mixed integer programmiing, I: Characterization results. *Discrete Applied Mathematics 17*, 3 (1987), 223–243.

[104] Jeroslow, R. G., and Lowe, J. K. Modelling with integer variables. In *Mathematical Programming at Oberwolfach II*. Springer, 1984, pp. 167–184.

[105] Josz, C., Maeght, J., Panciatici, P., and Gilbert, J. C. Application of the moment-SOS approach to global optimization of the OPF problem. *IEEE Transactions on Power Systems 30*, 1 (2015), 463–470.

[106] Kamath, A., and Karmarkar, N. A continuous approach to compute upper bounds in quadratic maximization problems with integer constraints. In *Recent Advances in Global Optimization* (1992), Princeton University Press, pp. 125–140.

[107] Kamath, A. P., and Karmarkar, N. K. An O(nL) iteration algorithm for computing bounds in quadratic optimization problems. In *Complexity in Numerical Optimization*. World Scientific, 1993, pp. 254–268.

[108] Karger, D., Motwani, R., and Sudan, M. Approximate graph coloring by semidefinite programming. *Journal of the ACM (JACM) 45*, 2 (1998), 246–265.

[109] Karush, W. Minima of functions of several variables with inequalities as side constraints. *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago* (1939).

[110] Kelley, Jr, J. E. The cutting-plane method for solving convex programs. *Journal of the society for Industrial and Applied Mathematics 8*, 4 (1960), 703–712.

[111] Kiwiel, K. C. Proximity control in bundle methods for convex nondifferentiable minimization. *Mathematical Programming 46*, 1-3 (1990), 105–122.

[112] Kocuk, B., Dey, S. S., and Sun, X. A. Strong SOCP relaxations for the optimal power flow problem. *Operations Research 64*, 6 (2016), 1177–1196.

[113] Kocuk, B., Dey, S. S., and Sun, X. A. New formulation and strong MISOCP relaxations for AC optimal transmission switching problem. *IEEE Transactions on Power Systems 32*, 6 (Nov 2017), 4161–4170.

[114] Konno, H., Kawadai, N., and Tuy, H. Cutting plane algorithms for nonlinear semi-definite programming problems with applications. *Journal of Global Optimization 25*, 2 (2003), 141–155.

[115] Krantz, S. G., and Parks, H. R. *A primer of real analytic functions*. Springer Science & Business Media, 2002.

[116] Krishnan, K., and Mitchell, J. E. *A linear programming approach to semidefinite programming problems*. PhD thesis, PhD thesis, Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, 2002.

[117] Kuhn, H. W., and Tucker, A. W. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (Berkeley, Calif., 1951), University of California Press, pp. 481–492.

[118] Land, A. H., and Doig, A. G. An automatic method of solving discrete programming problems. *Econometrica: Journal of the Econometric Society* (1960), 497–520.

[119] Lasserre, J. B. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization 11*, 3 (2001), 796–817.

[120] Lasserre, J. B. On representations of the feasible set in convex optimization. *Optimization Letters 4*, 1 (2010), 1–5.

[121] Lasserre, J. B. On convex optimization without convex representation. *Optimization Letters 5*, 4 (2011), 549–556.

[122] Lavaei, J., and Low, S. Zero duality gap in Optimal Power Flow problem. *IEEE Transactions on Power Systems 27*, 1 (feb. 2012), 92 –107.

[123] Lavaei, J., Tse, D., and Zhang, B. Geometry of power flows and optimization in distribution networks. *IEEE Transactions on Power Systems 29*, 2 (2014), 572–583.

[124] Lehmann, K., Grastien, A., and Hentenryck, P. V. AC-feasibility on tree networks is NP-hard. *IEEE Transactions on Power Systems 31*, 1 (Jan 2016), 798–801.

[125] Lesieutre, B. C., Molzahn, D. K., Borden, A. R., and DeMarco, C. L. Examining the limits of the application of semidefinite programming to power flow problems. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (2011), IEEE, pp. 1492–1499.

[126] Louca, R., Seiler, P., and Bitar, E. A rank minimization algorithm to enhance semidefinite relaxations of optimal power flow. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (2013), IEEE, pp. 1010–1020.

[127] Luo, Z.-Q., and Tseng, P. On the convergence of the coordinate descent method for convex differentiable minimization. *Journal of Optimization Theory and Applications 72*, 1 (1992), 7–35.

[128] Madani, R., Ashraphijuo, M., and Lavaei, J. Promises of conic relaxation for contingency-constrained optimal power flow problem. *IEEE Transactions on Power Systems* (2015).

[129] Madani, R., Kalbat, A., and Lavaei, J. A low-complexity parallelizable numerical algorithm for sparse semidefinite programming. *IEEE Transactions on Control of Network Systems* (2017).

[130] Malick, J., Povh, J., Rendl, F., and Wiegele, A. Regularization methods for semidefinite programming. *SIAM Journal on Optimization 20*, 1 (2009), 336–356.

[131] Mangasarian, O. L. Pseudo-convex functions. *Journal of the Society for Industrial and Applied Mathematics, Series A: Control 3*, 2 (1965), 281–290.

[132] Mareček, J., and Takáč, M. A low-rank coordinate-descent algorithm for semidefinite programming relaxations of optimal power flow. *Optimization Methods and Software 32*, 4 (2017), 849–871.

[133] Martin, D. The essence of invexity. *Journal of optimization Theory and Applications 47*, 1 (1985), 65–76.

[134] Martinet, B. Brève communication. régularisation d'inéquations variationnelles par approximations successives. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique 4*, R3 (1970), 154–158.

[135] Martínez-Legaz, J. E. What is invexity with respect to the same η? *Taiwanese Journal of Mathematics* (2009), 753–755.

[136] McCormick, G. Computability of global solutions to factorable nonconvex programs: Part I - convex underestimating problems. *Mathematical Programming 10* (1976), 146–175.

[137] Mendelson, B. *Introduction to topology.* Courier Corporation, 1990.

[138] Miao, Z., Fan, L., Aghamolki, H. G., and Zeng, B. Least squares estimation based SDP cuts for SOCP relaxation of AC OPF. *IEEE Transactions on Automatic Control 63*, 1 (2018), 241–248.

[139] Molzahn, D. K. Computing the feasible spaces of optimal power flow problems. *IEEE Transactions on Power Systems 32*, 6 (2017), 4752–4763.

[140] Molzahn, D. K., and Hiskens, I. A. Moment-based relaxation of the optimal power flow problem. In *Power Systems Computation Conference (PSCC), 2014* (2014), IEEE, pp. 1–7.

[141] Molzahn, D. K., and Hiskens, I. A. Sparsity-exploiting moment-based relaxations of the optimal power flow problem. *IEEE Transactions on Power Systems 30*, 6 (2015), 3168–3180.

[142] Nakata, K., Fujisawa, K., Fukuda, M., Kojima, M., and Murota, K. Exploiting sparsity in semidefinite programming via matrix completion II: Implementation and numerical results. *Mathematical Programming 95*, 2 (2003), 303–327.

[143] Nemhauser, G. L., and Wolsey, L. A. *Integer and Combinatorial Optimization. Interscience Series in Discrete Mathematics and Optimization.* John Wiley & Sons, 1988.

[144] Nesterov, Y. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization methods and software 9*, 1-3 (1998), 141–160.

[145] Nesterov, Y., and Nemirovskii, A. *Interior-point polynomial algorithms in convex programming.* SIAM, 1994.

[146] Nesterov, Y., and Nemirovsky, A. A general approach to polynomial-time algorithms design for convex programming. *Report, Central Economical and Mathematical Institute, USSR Academy of Sciences, Moscow* (1988).

[147] Nesterov, Y., and Nemirovsky, A. *Self-concordant functions and polynomial-time methods in convex programming.* USSR Academy of Sciences, Central Economic & Mathematic Institute, 1989.

[148] Nesterov, Y., and Nemirovsky, A. Conic formulation of a convex programming problem and duality. *Optimization Methods and Software 1*, 2 (1992), 95–115.

[149] Nesterov, Y. E., and Nemirovskii, A. S. Optimization over positive semidefinite matrices: Mathematical background and user's manual. *USSR Acad. Sci. Centr. Econ. & Math. Inst 32* (1990).

[150] Nocedal, J., and Wright, S. J. *Numerical optimization 2nd edition*. Springer, 2006.

[151] Norden, A. P. Theory of surfaces (in Russian). *Gostekhizdat, Moscow 8* (1956).

[152] O'Neill, R., Baldick, R., Helman, U., Rothkopf, M., and Stewart, W., J. Dispatchable transmission in RTO markets. *IEEE Transactions on Power Systems 20*, 1 (2005), 171–179.

[153] Oustry, F. A second-order bundle method to minimize the maximum eigenvalue function. *Mathematical Programming 89*, 1 (2000), 1–33.

[154] Padhy, N. P. Unit commitment – a bibliographical survey. *IEEE Transactions on Power Systems 19*, 2 (2004), 1196–1205.

[155] Pardalos, P. M., and Schnitger, G. Checking local optimality in constrained quadratic programming is NP-hard. *Operations Research Letters 7*, 1 (1988), 33–35.

[156] Peschon, J., Bree, D. W., and Hajdu, L. P. Optimal power-flow solutions for power system planning. *Proceedings of the IEEE 60*, 1 (1972), 64–70.

[157] Phan, D. T. Lagrangian duality and branch-and-bound algorithms for optimal power flow. *Operations Research 60*, 2 (2012), 275–285.

[158] Potluri, T., and Hedman, K. W. Impacts of topology control on the ACOPF. In *Power and Energy Society General Meeting, 2012 IEEE* (2012), IEEE, pp. 1–7.

[159] Povh, J., Rendl, F., and Wiegele, A. A boundary point method to solve semidefinite programs. *Computing 78*, 3 (2006), 277–286.

[160] Raman, R., and Grossmann, I. E. Modelling and computational techniques for logic based integer programming. *Computers & Chemical Engineering 18*, 7 (1994), 563–578.

[161] Rao, B. V., Kumar, G. N., Priya, M. R., and Sobhan, P. Optimal power flow by Newton method for reduction of operating cost with SVC models. In *2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies* (2009), IEEE, pp. 468–470.

[162] Rau, N. S. Issues in the path toward an RTO and standard markets. *IEEE Transactions on Power Systems 18*, 2 (2003), 435–443.

[163] Rau, N. S. *Optimization principles: practical applications to the operation and markets of the electric power industry*, vol. 16. John Wiley & Sons, 2003.

[164] Riccietti, E., Bellavia, S., and Sello, S. Sequential linear programming and particle swarm optimization for the optimization of energy districts. *Engineering Optimization 51*, 1 (2019), 84–100.

[165] Robertson, N., and Seymour, P. D. Graph minors. III. Planar tree-width. *Journal of Combinatorial Theory, Series B 36*, 1 (1984), 49–64.

[166] Robinson, S. M. A quadratically-convergent algorithm for general nonlinear programming problems. *Mathematical Programming 3*, 1 (1972), 145–156.

[167] Rockafellar, R. T. *Convex analysis.* Princeton University Press, 2015.

[168] Savelsbergh, M. A branch-and-price algorithm for the generalized assignment problem. *Operations Research 45*, 6 (1997), 831–841.

[169] Schramm, H., and Zowe, J. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. *SIAM Journal on Optimization 2*, 1 (1992), 121–152.

[170] Shapiro, A. First and second order analysis of nonlinear semidefinite programs. *Mathematical Programming 77*, 1 (1997), 301–320.

[171] Simmons, G. F. *Introduction to topology and modern analysis.* Tokyo, 1963.

[172] Smith, E. M., and Pantelides, C. C. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex minlps. *Computers & Chemical Engineering 23*, 4-5 (1999), 457–478.

[173] Sojoudi, S., and Lavaei, J. Physics of power networks makes hard optimization problems easy to solve. In *Power and Energy Society General Meeting, 2012 IEEE* (2012), IEEE, pp. 1–8.

[174] Sojoudi, S., Madani, R., and Lavaei, J. Low-rank solution of convex relaxation for optimal power flow problem. In *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)* (2013), IEEE, pp. 636–641.

[175] Stott, B., and Hobson, E. Power system security control calculations using linear programming, part I. *IEEE Transactions on Power Apparatus and Systems*, 5 (1978), 1713–1720.

[176] Stott, B., and Hobson, E. Power system security control calculations using linear programming, part II. *IEEE Transactions on Power Apparatus and Systems*, 5 (1978), 1721–1731.

[177] Stott, B., Jardim, J., and Alsaç, O. DC power flow revisited. *IEEE Transactions on Power Systems 24*, 3 (2009), 1290–1300.

[178] Stubbs, R. A., and Mehrotra, S. A branch-and-cut method for 0-1 mixed convex programming. *Mathematical Programming 86*, 3 (1999), 515–532.

[179] Tawarmalani, M., and Sahinidis, N. V. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming 103*, 2 (2005), 225–249.

[180] Tawarmalani, M., Sahinidis, N. V., and Sahinidis, N. *Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications*, vol. 65. Springer Science & Business Media, 2002.

[181] Taylor, J. A., and Hover, F. S. Linear relaxations for transmission system planning. *IEEE Transactions on Power Systems 26*, 4 (2011), 2533–2538.

[182] Telgen, J. Identifying redundant constraints and implicit equalities in systems of linear constraints. *Management Science 29*, 10 (1983), 1209–1222.

[183] Torres, G. L., and Quintana, V. H. An interior-point method for nonlinear optimal power flow using voltage rectangular coordinates. *IEEE Transactions on Power Systems 13*, 4 (1998), 1211–1218.

[184] Vandenberghe, L., and Boyd, S. Semidefinite programming. *SIAM Review 38*, 1 (1996), 49–95.

[185] Veinott Jr, A. F. The supporting hyperplane method for unimodal programming. *Operations Research 15*, 1 (1967), 147–152.

[186] Verma, A. *Power grid security analysis: An optimization approach.* PhD thesis, Columbia University, 2009.

[187] Wächter, A., and Biegler, L. T. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming 106*, 1 (2006), 25–57.

[188] Wang, Z., Fang, S.-C., and Xing, W. On constraint qualifications: motivation, design and inter-relations. *Journal of Industrial and Management Optimization 9*, 4 (2013), 983–1001.

[189] Wen, Z., Goldfarb, D., and Yin, W. Alternating direction augmented Lagrangian methods for semidefinite programming. *Mathematical Programming Computation 2*, 3-4 (2010), 203–230.

[190] Wolfe, P. Accelerating the cutting plane method for nonlinear programming. *Journal of the Society for Industrial and Applied Mathematics 9*, 3 (1961), 481–488.

[191] Wolkowicz, H., Saigal, R., and Vandenberghe, L. *Handbook of semidefinite programming: theory, algorithms, and applications*, vol. 27. Springer Science & Business Media, 2012.

[192] Wrede, R. C., and Spiegel, M. R. *Advanced calculus.* McGraw-Hill, 2010.

[193] Xia, Y., Wang, S., and Sheu, R.-L. S-lemma with equality and its applications. *Mathematical Programming 156*, 1 (Mar 2016), 513–547.

[194] Yu, Z. Solving semidefinite programming problems via alternating direction methods. *Journal of Computational and Applied Mathematics 193*, 2 (2006), 437–445.

[195] Zhang, B., and Tse, D. Geometry of feasible injection region of power networks. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (2011), IEEE, pp. 1508–1515.

[196] Zhang, W., Li, F., and Tolbert, L. M. Review of reactive power planning: objectives, constraints, and algorithms. *IEEE Transactions on Power Systems 22*, 4 (2007), 2177–2186.

[197] Zhao, X.-Y., Sun, D., and Toh, K.-C. A Newton-CG augmented Lagrangian method for semidefinite programming. *SIAM Journal on Optimization 20*, 4 (2010), 1737–1765.